User Manual
EUROM (SAA5350)

Issue 1 (Provisional) February 1986

TABLE OF CONTENTS

Paragraph No.		Page No.
	Section 1 **General Description**	
1.1	Introduction	5
	Using This Manual	5
1.2	Main Features	5
1.3	Operating Modes - General	7
	Stack Architecture	7
	Explicit Fill	7
	80 Characters/Row	7
1.4	Timing Chain	8
1.5	Synchronisation Modes	8
1.6	Character Generation	10
1.7	Scroll Map	11
1.8	Row Attributes	12
	Disable Ram Access (DRA)	12
	Disable Text Display (DTD)	12
	Disable Double Height (DDH)	12
	Row Address	13
	Row Reveal	13
	Zoom	13
	Cursor Row	13
	Screen Colours	13
1.9	Colour Map and D/A Converters	13
	DRCS Colour Look-up Table	15
	D-A Connecters	15
1.10	Character Table	15
1.11	Display Attributes	17
	Foreground Colour	17
	Background Colour	17
	Screen Colour	17
	Flash	17
	Character Size	18
	Lining	19
	Conceal	19
	Invert	19
	Window/Box	19
	White Button	20
	Section 2 **Operating Modes**	
2.1	Introduction	21
2.2	Stack Mode (40 Characters/Row)	21
	Stack Coding	21
	Lock Bit	23
	Cursor (Stack Mode)	23
	Flash Coding (Stack Mode)	24
	Memory Addressing (Stack Mode)	26
	Character Addresses	28
	Attribute Addresses	28
	Screen Colour Addresses	28

TABLE OF CONTENTS (Continued)

Paragraph No. Page No.

Section 2
Operating Modes (Continued)

2.3 Explicit Fill Mode (40 Characters/Row) 29
Cursor (Explicit Fill Mode, 40 Characters/Row) 29
Flash Coding (Explicit Fill Mode) 29
Memory Addressing (Explicit Fill Mode) 29
2.4 80 Characters/Row Mode 34
Cursor (80 Characters/Row Mode) 34
Character/Attribute Memory Addressing (80 Characters/Row Mode) 34
Window/Box Enable (80 Characters/Row Mode) 37
2.5 Dot Data Addressing 37

Section 3
Dynamically Redefinable Character Sets (DRCS)

3.1 Introduction 38
3.2 DRCS Character Cells 38
3.3 Interpretation of Data Bits 40
3.4 DRCS Addressing (40 Characters/Row) 40
3.5 DRCS Addressing (80 Characters/Row) 42
3.6 DRCS Mode Coding 43
3.7 Full Field DRCS 64
Bit Map Example 65

Section 4
EUROM Register Map

4.1 Use of Register Map 67
4.2 Register Map Conventions 67
4.3 Register 0 (R0) - Sync Register 69
4.4 Register 1 (R1) - Test Register 73
4.5 Register 2 (R2) - DMA Warning Time Register 73
4.6 Register 3 (R3) - Display Command Register 1 75
4.7 Register 4 (R4) - Display Command Register 2 78
4.8 Register 5 (R5) - Display Start Register 79
4.9 Register 6 (R6) - Display Page Chapter Pointer 80
4.10 Register 7 (R7) - DRCS Chapter 'A' Pointer 80
4.11 Register 8 (R8) - DRCS Chapter 'B' Pointer 80
4.12 Register 9 (R9) - Cursor Column 81
4.13 Register 10 (R10) - Cursor Character Table and Foreground Colour 81
4.14 Register 11 (R11) - Cursor Flash Enable, Conceal and Background Colour 81
4.15 Register 12 (R12) - Cursor Lining and Character Code 82
4.16 Register 13 (R13) - DRCS - CLUT 82
4.17 Register 14 (R14) - Colour Map Address 83
4.18 Register 15 (R15) - Colour Map Data (RED) 83
4.19 Register 16 (R16) - Colour Map Data (GREEN) 83
4.20 Register 17 (R17) - Colour Map Data (BLUE) 84
4.21 Registers R18 to R32 - Unused 84
4.22 Registers R33 to R58 (R33 to R58) - Scroll Map 84

Paragraph No. Page No.

Section 5
EUROM Application Details

5.1 Pinning 88
5.2 Microprocessor and RAM Interfacing 89
EUROM Access to Display Memory 90
Display Memory Transfer Timing (Bus Request) 91
Microprocessor Access to EUROM's Register Map 93
8-bit Microprocessor Operation 95
Disconnected Systems 96
5.3 Cycle Interleaving 97
5.4 Scroll Map Availability 97
5.5 Synchronising EUROM to an external video source 99
5.6 Power-On Reset 100

REFERENCES

References 1-4 103

Appendix 1
CEPT Videotex

1.1 CEPT Videotex 104
Basic Alphamosaic System 104
'Alpha-DRCS' System 104
Alphageometric System 104
Alphaphotographic System 105

INDEX

A-Z 106
Numbers 110

REGISTER MAP

Table 4.1 Repeated from page 68 111

SECTION 1

GENERAL DESCRIPTION

1.1 INTRODUCTION

EUROM is a 40-pin single-chip VLSI NMOS crt controller, type SAA5350, designed to meet the requirements of the many videotex services now being developed and introduced throughout the world. Its main areas of application are in CEPT-standard terminals, and also in 80-characters per row displays and bit-map displays. EUROM handles all the presentation level aspects required by the CEPT 'alpha-DRCS' videotex terminal Profile 1 (as used, for example, by the Bildschirmtext Service in West Germany) but it also offers many additional features. With EUROM, only minimal hardware is required to produce an inexpensive terminal: in the simplest configuration, little more than a microprocessor and 4 Kbytes of display memory, mounted on a small printed-circuit board.

USING THIS MANUAL

The 'Contents' at the front of this Manual indicates the order in which information about the EUROM integrated circuit SAA5350 is given in the Manual.

The reader who is already familiar with devices of this type may wish to read the 'Main Features' and 'Operating Modes' in Sections 1.1. and 1.2 and then consult the 'Application Details' in Section 5 and the published Data Sheet.

While the reader is examining details in the major portion of the Manual he may wish to consult the Register Map of the SAA5350 frequently. This is given in Section 4.2 and also as a 'pull-out sheet' at the rear of this Manual.

1.2 MAIN FEATURES

The main features of EUROM are:

(1) 40/80 character by 1-to-25 row display.

(2) 512 alphanumeric and graphical characters on-chip.

(3) Dynamically Redefinable Character capability (DRCS).

(4) Interfaces with 8/16-bit microprocessors with optional direct memory access.

(5) On-chip scroll map eliminates the need for massive data transfer when scrolling.

(6) On-chip colour map RAM, and three non-linear D/A converters to compensate for CRT non-linearity.

(7) Memory interface capable of supporting multi-page terminals. EUROM can access up to 128 Kbytes of display memory.

(8) A 25th row which is displayed by EUROM can be used for local status messages in BTX applications.

(9) Three synchronisation modes:

(a) Stand-alone - using an external 6 MHz crystal with an on-chip oscillator.
(b) Simple slave directly synchronised from source of text composite sync.
(c) Phase-locked slave enabling the synchronisation of video and text displays, i.e. VCR/VLP or TV video with text overlay.

(10) On-chip timing with composite sync output.

(11) Zoom - this allows the height of any group of rows to be increased for enhanced legibility.

(12) Full Field DRCS - for alphageometric and similar applications, a bit-map display is desirable, where each pixel on the screen corresponds to a location in memory. EUROM implements this indirectly by expanding the DRCS character repertoire so that the entire defined display area can be covered with fully random data.

A basic block diagram of EUROM is shown in Figure 1.1. Generally, character and attribute data is fetched from an external memory, processed by the row buffer fill logic, and then fed into one half of the dual display row buffer. The data fetch process takes place during one or two (depending on display mode) line flyback periods per row, and since time is required to complete the fill, the other half of the dual row buffer is used for display (see Section 2.1). The two halves of the row buffer exchange functions on alternate rows. Each half of the row buffer holds the 40 columns of 32 bits, or 80 columns of 16 bits, required to define explicitly every character in a row. The addresser is used for row buffer filling and for fetching screen colours; during the display time it is also used for addressing DRCS characters. (The organisation of RAM is described in section 5.2, and Row Buffer filling is described in Sections 1.3, 2.1 and 4.2).

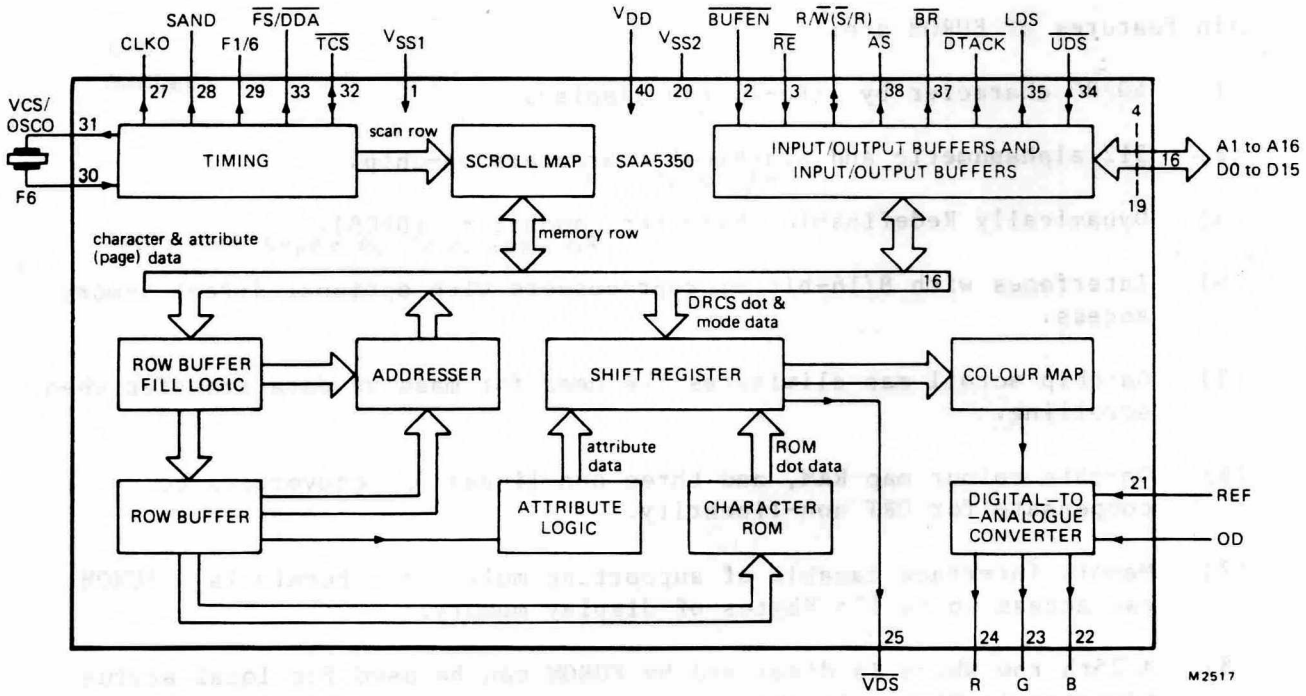


Figure 1.1 Simplified block diagram

1.3 OPERATING MODES - GENERAL

The fundamental operation of EUROM uses Stack Architecture for the storage of characters and attributes in memory. However, Explicit Fill modes without stack coding may also be used.

Stack Architecture

To fully implement all the functions in EUROM, 32 bits per character location are necessary as follows.

7	Bits Character Code
3	Bits Character Table
5	Bits Foreground Colour
5	Bits Background Colour
5	Bits Flash
3	Bits Size (D.Ht., Top/Bot., D.Width)
1	Bit Lining (Underline or Mosaic Separation)
1	Bit Conceal
1	Bit Invert
1	Bit Window/Box
32	Bits per character position

To display a full screen of 25 rows of 40 characters, therefore, 4 Kbytes of external memory are required. To reduce the amount of memory required, attributes in EUROM are coded using Stack Architecture. This exploits the natural redundancy of normal text by allocating memory dynamically, thus allowing the external memory to be reduced to 2 Kbytes per screen (see Stack Coding in Section 2.2). One Row Buffer fill occurs per row (see Sections 2.1 and 4.2). A Cursor can be generated by the EUROM Cursor Register in this mode.

If DRCS storage is to be used, then an additional 2 Kbytes of memory is required (total 4 Kbytes).

Explicit Fill

In Explicit Fill mode, the page memory is not stack coded, and no processing is carried out during the Row Buffer Fill operation. Data from the memory is transferred directly to the row buffer.

Since there is then an explicit representation of all the attributes at every character location, there is no limit to the number of attribute changes on a given row. However, this mode requires a larger amount of external RAM (6 Kbytes/page including DRCS memory). Also, enlarged characters are not checked, so the rules concerning the size attributes must be implemented in software. Two Row Buffer fills occur per row (see Sections 2.1 and 4.2). Any Cursor must be generated in software because the EUROM Cursor Registers become ineffective in this mode.

80-Characters/Row

The 80-Character mode is also an Explicit Fill mode without stack coding. No additional circuitry is required; the row buffer is effectively rearranged as 80 16-bit words, each containing:

- 8 character bits
- 3 foreground colour bits
- 3 background colour bits
- 1 underline bit
- 1 flash bit

Dot data is fetched from external memory in the same way that DRCS data is retrieved. All characters are displayed as a 6 x 10 dot matrix, with both one and two bits/dot modes available. In the one bit/dot mode, the external dot memory need only be eight bits wide. When using 10 lines/row, 204 different character matrices may be stored in a 2K8 memory. Two Row Buffer fills occur per row (see Sections 2.1 and 4.2).

The flash mode incorporates colour table flash. For maximum flexibility of display the foreground and background colours are applied to different areas of the colour map. Any Cursor must be generated on software in this mode since the Cursor Registers become ineffective.

1.4 TIMING CHAIN

EUROM timing is operated from an external 6 MHz clock or an on-chip fixed frequency crystal oscillator, see Figure 1.1. The basic format is 40 characters per row, 25 rows per page and 10 video lines per row. The normal transmission format generally uses up to 24 rows, and the 25th row can be used for local status display if required.

An extra bit in the Scroll Map (1 bit per row) enables the number and position of displayed rows on a page to be controlled. In this way, the number of rows of display can be limited to less than 25, and the resulting display can be placed at the top, or centre, or bottom of the screen as required.

The device also has a 12 lines per row mode. The two extra lines per row are added symmetrically and contain background colour only for ROM-based alphanumeric characters. DRCS characters, mosaics ('block' and 'smooth') and line drawing characters, however, occupy all 12 lines in this mode.

The display is generated to the normal 625 line/50 Hz scanning standard, interlaced or non-interlaced. A composite sync output is provided for conventional timebases. A clock output at 1 MHz or 6 MHz is available for driving other videotex devices such as the SAA5070, and a 12 MHz clock is available for dot synchronisation of hard copy etc.

A Defined Display Area ($\overline{\text{DDA}}$) output timing signal simplifies the application of external peripherals such as a light pen and hard copy printer. This signal is nominally coincident with the character information, corresponding to the 40 (80) characters and up to 25 rows.

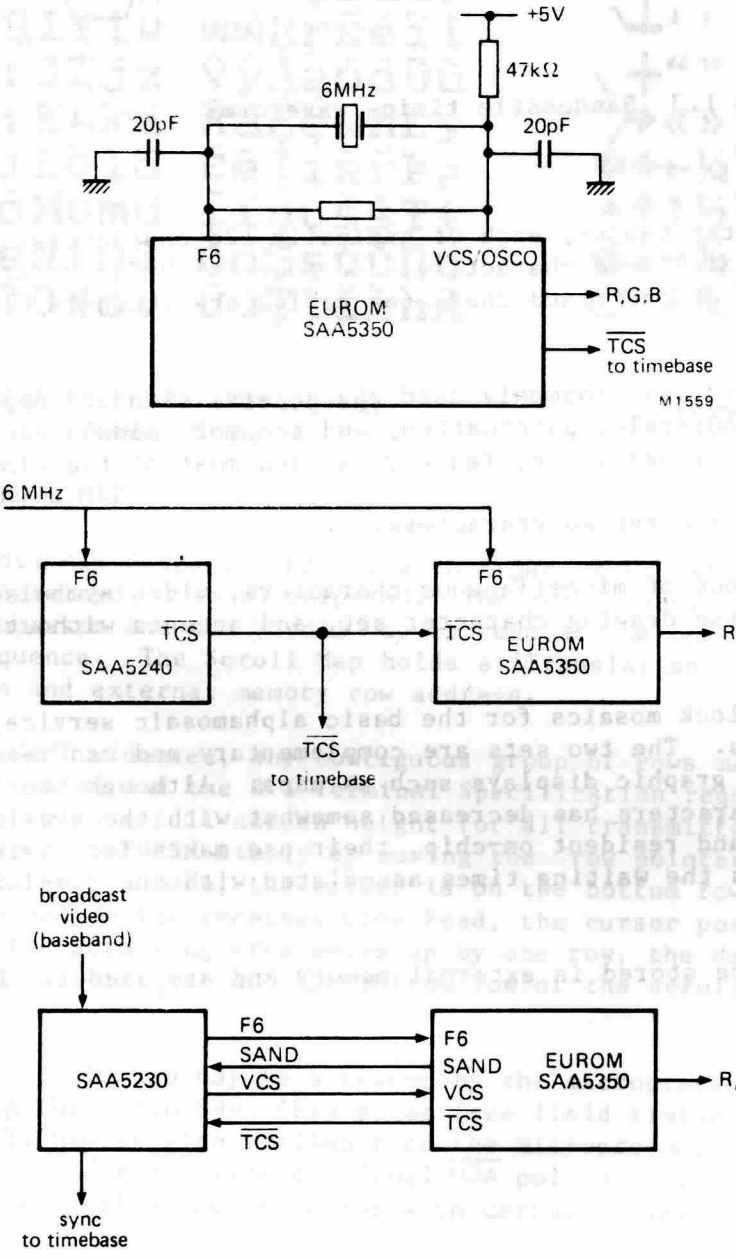
1.5 SYNCHRONISATION MODES

There are three basic synchronisation modes as shown in Figure 1.2.

(1) Stand-alone. The on-chip fixed frequency oscillator works in conjunction with an external 6 MHz crystal. All timing is generated internally.

(2) Simple Slave. EUROM can sync directly to another device, such as the Text Composite Sync signal $\overline{\text{TCS}}$ from a Computer Controlled Teletext (CCT, SAA5240) or another EUROM as shown in Figure 1.2. EUROM's horizontal counter is reset by the falling edge of $\overline{\text{TCS}}$. So that line-to-line jitter does not occur, the horizontal counter is not reset if $\overline{\text{TCS}}$ falls within a specified 250 ns period. This period follows a period of nearly 64 µs which starts from the falling edge of one $\overline{\text{TCS}}$ signal. Field synchronisation is achieved with EUROM's internal field sync separator.

(3) Phase-Locked Slave. An important application of CEPT videotex (such as Bildschirmtext in Germany) involves the display of a page of text with a 'window' onto part of a still video picture derived from a Video Long Player (VLP) laser-read disc. This mode requires that the 6 MHz system clock be phase locked to the video from a VLP and that the horizontal and vertical timing chains are properly synchronised. In Figure 1.2c, part of a SAA5230 forms the phase-locked loop. 'Sandcastle' feedback from EUROM completes the phase-locked loop and determines the horizontal display phase (see Figure 1.3 for timing waveforms). The vertical display is synchronised by an on-chip field sync separator fed from the SAA5230 Video Composite Sync output. This mode is also suitable for broadcast applications. More details of this arrangement can be found in Section 5.5.



(a) Stand-alone

(b) Simple slave (direct sync)

(c) Phase-locked slave (indirect sync)

Figure 1.2 Synchronisation modes

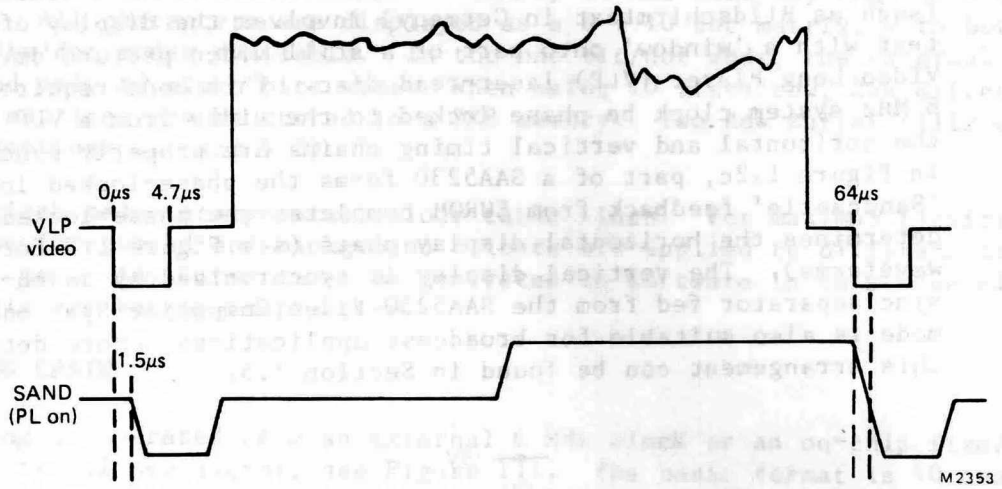


Figure 1.3 Sandcastle timing waveforms

1.6 CHARACTER GENERATION

EUROM supports eight character tables, each of nominally 128 characters. Four are in on-chip ROM and contain fixed characters, and four for DRCS are stored in external RAM. The contents of the fixed character tables are shown in Figures 1.4 and 1.5.

Table 0 contains the 128 most commonly used characters: standard upper and lower-case Roman alphabet, numerals, punctuation, and the more common accented characters. In normal text transmission, Table 0 is used most of the time.

Table 1 contains further accented characters.

Table 2 contains a number of miscellaneous characters, video symbols, mathematical symbols, the line drawing character set, and accents without associated characters.

Table 3 contains the block mosaics for the basic alphamosaic service, together with smooth mosaics. The two sets are complementary and can readily be combined to create pleasing graphic displays such as maps. Although the editorial need for these characters has decreased somewhat with the availability of DRCS, being pre-defined and resident on-chip, their use makes for savings in transmission time and avoids the waiting times associated with the downloading of DRCS characters.

Tables 4, 5, 6 and 7 are stored in external memory and are used for DRCS.

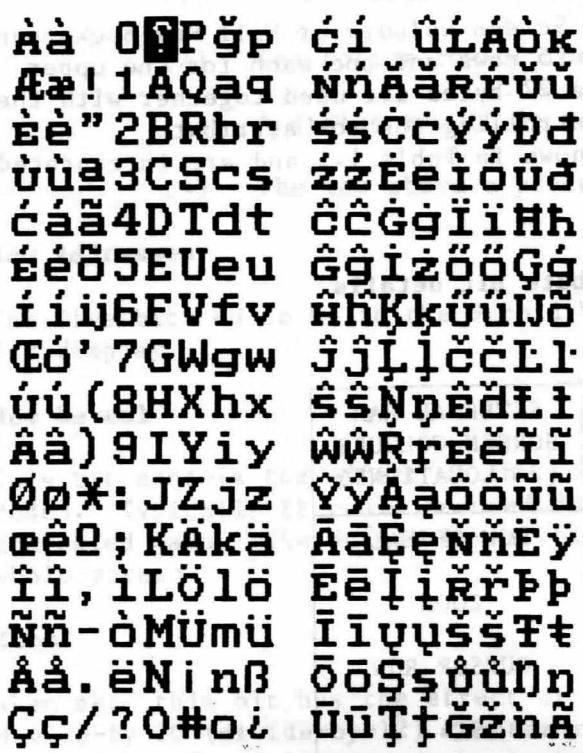


Figure 1.4 On-chip characters: Tables 0 and 1

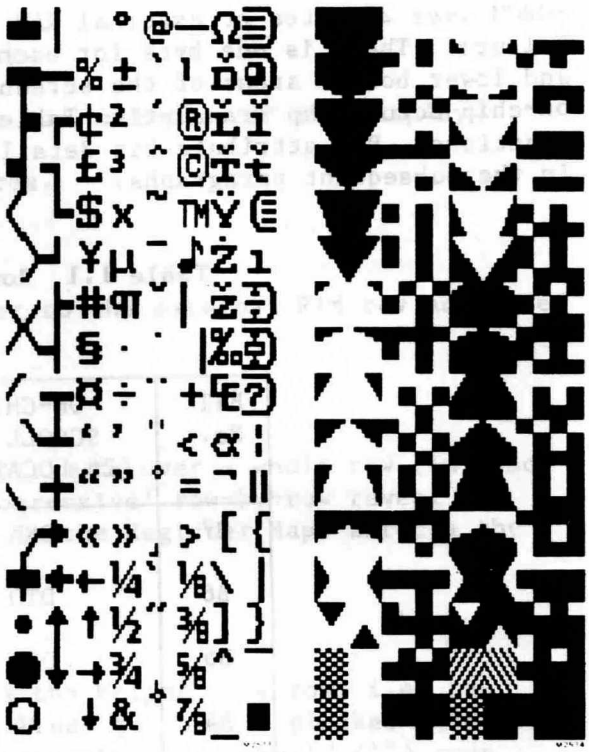


Figure 1.5 On-chip characters: Tables 2 and 3

1.7 SCROLL MAP

The Scroll Map which is indicated in Figure 1.1 comprises an area of 26 x 8-bit words contained in an on-chip RAM. The scroll words are defined by the microprocessor and the Scroll Map is read at the beginning of each Row Buffer Fill sequence. The Scroll Map holds a 'Translation Table' between screen position and external memory row address.

In CEPT videotex, any contiguous group of rows may be defined to be a scrolling area and the BTX Terminal Specification requires that scroll must be possible up to a full screen height for all transmitted characters. This effect is achieved in EUROM simply by moving the "row pointers" which are held in Scroll Map. If, for example, the cursor is on the bottom row of a scrolling area and the external device receives Line Feed, the cursor position is unaffected, the data in the scrolling area moves up by one row, the data in the top row of the scrolling areais lost and the bottom row of the scrolling area is filled with spaces.

The Scroll Map may be accessed by the microprocessor during the period following the Field Sync ($\overline{FS}$) pulse (see field timing waveforms Section 5.4). The Scroll Map is also available to the microprocessor from the final line of display in a field, (i.e. the final $\overline{DDA}$ pulse) up to $\overline{FS}$. It should be noted that this latter period may not exist with certain combinations of Zoom and 12 lines per row.

1.8 ROW ATTRIBUTES AND ROW CONTROL

EUROM uses 27 bytes of external RAM to hold Screen Colours or Full Row Background Colours. There is one byte for each of the 25 rows and one each for the upper and lower border areas of the screen. These 27 bytes are used together with the on-chip Scroll Map Translation Table for controlling the row attribute functions. Row attribute bit details are shown in Table 1.1 and are interpreted in the subsequent paragraphs.

Table 1.1 Row attribute bit details

BIT No.	ON-CHIP SCROLL MAP (26 LOCATIONS)	EXTERNAL RAM SCREEN COLOURS (27 LOCATIONS)
B7	DRA	ROW REVEAL
B6	DTD	ZOOM
B5	DDH	CURSOR ROW
B4	R4	SC4
B3	R3	SC3
B2	R2	SC2
B1	R1	SC1
B0	R0	SC0

Disable Ram Access (DRA)

This bit may be used to inhibit row buffer filling, screen colour fetching or DRCS processing.

Disable Text Display (DTD)

May be used to control the number and position of rows on a page. (DTD therefore defines the vertical component of DDA, the Defined Display Area.) In conjunction with DRA, this bit may also be used to control EUROM so that one or two rows are displayed with no RAM access required.

Disable Double Height (DDH)

Considerable software saving can be achieved by inhibiting double height over a whole row in hardware. This bit enables/inhibits the double height attribute over a whole row. Double size characters display double width.

Note: This satisfies the requirements of a CEPT videotex service where the origin of a double height or double size character is not allowed to occur in any of the following positions:

(1) A row above a scroll area.

(2) The bottom row of the display.

(3) A status row if it is at the top or bottom of the display.

(4) The row above a protected area.

Row Addresses

The five bits R0 to R4 in the Scroll Map refer to the external RAM row addressed for display.

Row Reveal

This bit cancels the effect of the Conceal attribute over a whole row (in Stack Mode). Typically it could be used for a 'progressive' row-by-row reveal of concealed data. The Screen Reveal bit (R4/2 in the Register Map) affects the whole screen.

Zoom

When set, this bit has the effect of doubling the height of a row, i.e. the divide-by-10 (divide-by-12) counter relating lines to rows is clocked at half line rate. The following rows on the page are shifted down by 10 (12) scan lines. The effect is independent of any double height attribute and so any double-height characters become quadruple-height. Typically it will be used in conjunction with the Scroll Map to give Top/Bottom Half Page Expand. Other effects are of course possible.

Cursor Row

The cursor row bit flags on which row the cursor is to be displayed. This bit is controlled by the microprocessor, so cursor on/off, cursor flash rate and mark-space ratio, even multiple cursors are all software controllable.

Screen Colours

The remaining five bits (SC0 to SC4) in the external RAM screen colour locations are the values for the screen colours. If 25 rows of text are displayed, 27 screen colours may be required - one for each row and one each for the upper and lower border areas of the screen.

When scrolling text, the screen colours remain 'in place', that is, they do not move with the moving text.

1.9 COLOUR MAP AND D-A CONVERTERS

The on-chip Colour Map and D-A (Digital-to-Analogue) converters considerably simplify the external circuitry. The Colour Map RAM contains 32 x 12-bit words that are loaded by the microprocessor and read out at pixel rate. Typical contents of the Colour Map are shown in Table 1.2. Attention is drawn to the Notes 1 to 3 below Table 1.2.

Table 1.2 Typical Colour Map contents for videotex applications

	COLOUR MAP ENTRY NUMBERS	NORMAL COLOUR	NORMAL DATA CONTENTS B	G	R
Colour Look-Up Table 0	0	Transparent	0000	0000	0000
	1	Reduced intensity Red	0000	0000	0111
	2	Reduced intensity Green	0000	0111	0000
	3	Reduced intensity Yellow	0000	0111	0111
	4	Reduced intensity Blue	0111	0000	0000
	5	Reduced intensity Magenta	0111	0000	0111
	6	Reduced intensity Cyan	0111	0111	0000
	7	Grey	0111	0111	0111
Colour Look-Up Table 1	8	Black	0000	0000	0000
	9	Red	0000	0000	1111
	10	Green	0000	1111	0000
	11	Yellow	0000	1111	1111
	12	Blue	1111	0000	0000
	13	Magenta	1111	0000	1111
	14	Cyan	1111	1111	0000
	15	White	1111	1111	1111
Colour Look-Up Table 2	16 17 18 19 20 21 22 23	Redefinable 16 Colours	Default as Colour Look-Up Table 1		
Colour Look-Up Table 3	24 25 26 27 28 29 30 31		Default as Colour Look-Up Table 1		

NOTES

1. EUROM differs slightly from BTX in that the Bildschirmtext Videotex Specification makes Table 0 equivalent to CLUT2, Table 1 equivalent to CLUT1, Table 2 equivalent to CLUT4, and Table 3 equivalent to CLUT3.

2. The contents of Colour Map Location 0 should normally be 0000 0000 0000 because this location is used as output during line blanking and field blanking. Some a.c.-coupled monitors with clamped black-level use this output during line blanking as a black level reference.

3. The data shown in the table above is defined by the Bildschirmtext standard, but, apart from the desirability of having all 0s in CMAP Location 0 it is otherwise possible to have any of the 4096 colours of the full palette in any of the locations.

DRCS Colour Look-Up Table

A DRCS Colour Look-up Table (DCLUT) is also included for use when operating with 2 bits/pixel colour DRCS. This acts as a small RAM that maps the four combinations of two bits into any four of the 32 locations in the colour map (see Section 3.2).

In the 4-bits/pixel colour DRCS mode, the four bits map either the first 16 locations of the Colour Map or the second 16 locations of the Colour Map depending on the Palette bit in the EUROM Register 3 (R3/1, see Section 4.6).

D-A Converters

The Colour Map RAM output is split into three 4-bit groups each of which is fed to a non-linear (gamma corrected) D-A converter. The non-linearity of the D-A converter is intended to compensate for the non-linearity ('gamma') of the crt in such a way that the numbers fed to the D-A converter actually represent perceived brightness (as required by the CEPT specification). The three outputs (R, G and B) are fed to the output pins at a low impedance.

The peak-peak output amplitude is controlled by the external voltage applied to the Reference pin. Typically, two nominal amplitudes might be used as follows:

1 V pk-pk for direct interfacing to a standard colour decoder, and
2 V pk-pk for simple interfacing (using external buffer transistors) to a 1 V, 75 ohm video drive; see Video Interfaces Figure 1.6.

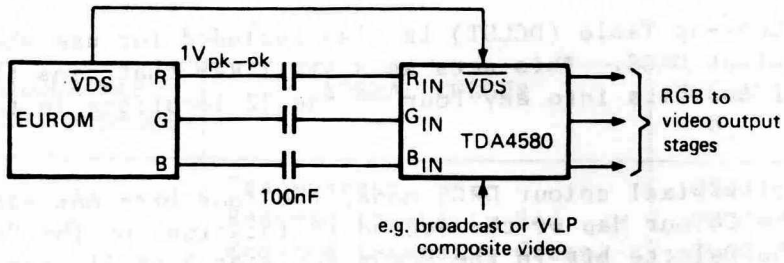
1.10 CHARACTER TABLE

The character table used for display is described by 3 bits, T2, T1 and T0 (implying character tables 0-7). Character tables 0-3 are for on-chip ROM (see Figures 1.4 and 1.5) and character tables 4-7 are for DRCS.

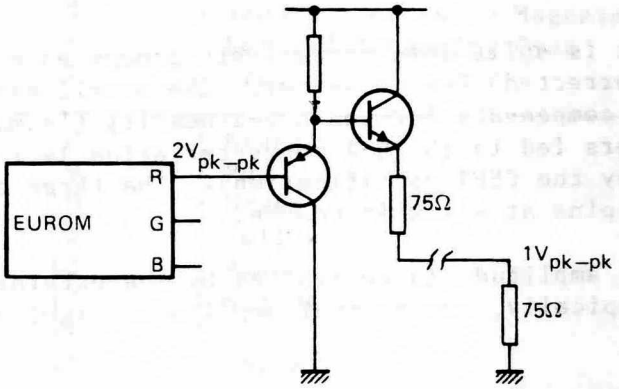
With DRCS, the individual bits have a further meaning as shown in Table 1.3. More detailed explanations of these bit-meanings are given in Section 3.3.

Table 1.3 DRCS character table

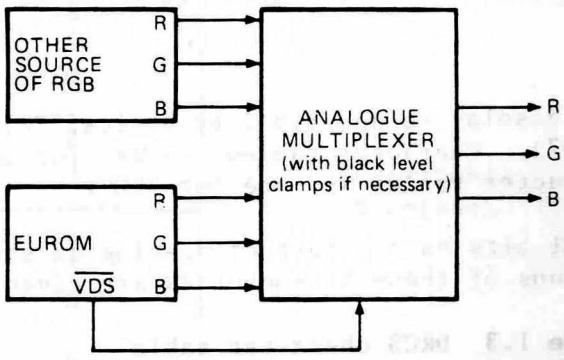
TABLE BIT	INTERPRETATION
T2	$\overline{\text{ROM}}$/DRCS. (Tables 0-3 are ROM, and Tables 4-7 are DRCS)
T1	$\overline{\text{DRCS Chapter Pointer A}}$/DRCS Chapter Pointer B (Tables 4,5 are DRCS Chapter A, and Tables 6,7 are DRCS Chapter B)
T0	$\overline{\text{DRCS memory cell left half}}$/DRCS memory cell right half (In 6x10x1 or 6x5x2 modes only)



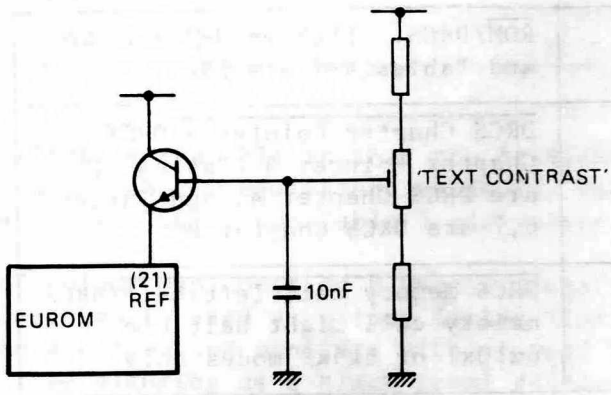
a) TDA4580 video interface



b) 1V 75Ω video interface



c) External analogue video multiplexed interface



d) Text contrast control M2516

Figure 1.6 Video interfaces

1.11 DISPLAY ATTRIBUTES

EUROM provides the following attributes for any character displayed:

Foreground Colour

Foreground colour is coded in five bits, implying a total of 32 colours. The five bits are used to address the colour map directly.

When a pixel is set to transparent (code 00000), the display colour pointer is set to the value of the screen colour at this location. If the screen colour attribute is also transparent, the underlying tv picture, if any, is unblanked (see Figure 1.7) by the video data switching (VDS) output from EUROM being set to logic 0 (LOW) for that pixel. When the tv picture is unblanked, the RGB outputs are taken from location 0 of the colour map.

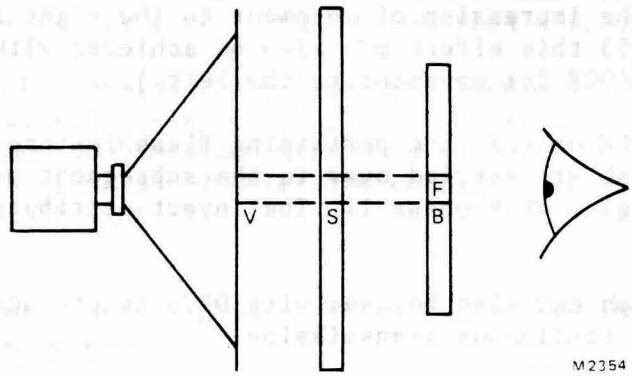


Figure 1.7 Interpretation of transparent

In Figure 1.7, foreground (F), background (B), and screen (S) colours are shown while V represents the video picture, visible when the screen and the character foreground or background colours are defined as transparent.

Background Colour

Background colour operates in exactly the same way as foreground colour, with five bits defining 31 colours and transparent.

Screen Colour

The colour of the screen may also be set to any one of the 31 colour map locations, or transparent. The screen is notionally divided into 27 areas corresponding to the 25 rows and the upper and low border areas. Each of these 27 areas can refer to a different location in the colour map.

Flash

There are three basic modes of flash:

(1) Flash to Background, where characters are displayed alternately in the prevailing foreground and the prevailing background colours.

(2) Colour Table Flash, where the active pixels alternate between two colours in the colour map. If the colour map is considered to consist of 4 Colour Look-up Tables (each of 8 locations), the effect of Colour Table Flash is to alternate the colour between the 'given' colour and the corresponding position in the adjacent Colour Look-Up Table (CLUT). This arrangement gives flashes from 0 to 1, 1 to 0, 2 to 3 and 3 to 2.

(3) Inverted Flash to Background (in anti-phase to mode 1).

In addition, there are two flash frequencies, fast (2 Hz) and slow (1 Hz), as well as four mark-space ratio phases. The mark-space ratio are as follows, with their phase identifications as used elsewhere.

50% mark-space ratio, "phase 0"
33% mark-space ratio, phase 1
33% mark-space ratio, phase 2
33% mark-space ratio, phase 3

If a row of identical objects are assigned subsequent flash phases 1, 2, 3, 1, 2, 3, etc, then the impression of movement to the right is achieved. In stack mode (see Section 2.5) this effect may also be achieved with a single implicit flash attribute ICF (DCF for movement to the left.)

In a field of ICF or DCF, the pertaining flash features such as anti-phase and colour table flash are carried over to the subsequent positions. A field of ICF or DCF always begins with phase 1. The Invert attribute charges the polarity of the flash.

Three phase flash can also be used with DRCS to produce dynamic displays without the need for continuous transmission.

Character Size

EUROM provides Normal Size, Double Size, Double Height and Double Width characters. In Stack Mode double-height characters extended downwards to the row below, and double-width characters extended to the right. Within the CEPT rules there are, however, certain restrictions (basically that part-characters may not be displayed).

In Explicit Fill Mode, the use of the Double Width attribute can be considered as the use of character columns in pairs, each with a left member of the pair and the right member of the pair. With the exception of the Double Width attribute itself, all character code bits and attribute bits are active in both the left and right members of the pairs. These bits in the left member of the pair would therefore normally be duplicated by software in the right member of the pair for each double-width character. The double-width attribute itself is "don't care" at the right hand position. Double-width characters thus extended to the right.

Within CEPT Videotex, the display of a character at position 'x' in a set of character positions A, B, C, D and x as shown, is governed by the four rules given below:

B	C	D
A	x	

(1) If position A is not obscured and contains the origin of a Double Width or Double Size character, character and attribute display data for position 'x' is taken from A - position x is obscured by A.

(2) If position B is not obscured and contains the origin of a Double Size character, character and attribute display data for position 'x' is taken from B - position x is obscured by B.

(3) If position C is not obscured and contains the origin of a Double Height or Double Size character, character and attribute display data for position 'x' is taken from C - position x is obscured by C.

(4) If position D contains the origin of a Double Height or Double Size character, position x itself may not display as a Double Width character, any Double Width attribute bit will be interpreted as Normal Width (thus any double size character at position x would become double height only).

All the above rules concerning size are automatically implemented by EUROM when operating in Stack mode.

Lining

For alphanumeric characters, the Lining attribute underlines the character. For mosaics and line drawing characters, it separates the character into six blocks or sub-squares (mosaic separation).

Conceal

The Conceal attribute makes the foreground and background colours the same until a local reveal function is activated. The local reveal function can be applied either to the whole field or on a row-by-row basis, allowing progressive reveal in response to user interaction.

Invert

This attribute exchanges the foreground and background colours and is included for compatibility with Télétel transmission codes. It also applies to Flash, giving anti-phase instead of normal flash.

Window/Box

If the basic frame begins in tv mode (for mode selection see Table 4.4 or the Register 3 description in Section 4), this attribute superimposes a box containing text (Foreground and Background or Screen colours) on the tv picture. It is compatible with the Box function used in the basic alphamosaic teletext service.

If the basic frame begins in text mode, the attribute provides a Window. That is, it sets the screen colour to transparent at the character positions where it applies, so that the underlying tv picture is visible at pixels where foreground or background colours are transparent.

White Button

Various attributes and combinations of attributes can cause on-screen data to be obscured - double height/double width, conceal, foreground and background colours the same, etc. This effect can be negated by a user function, colloquially known as the 'White Button', which sets all the attributes to their default values without affecting the display memory contents. This function is implemented in EUROM by a microprocessor-defined register bit which is active in Stack Mode, Explicit Fill Mode, and 80 Characters/Row Mode. (See Operating Modes.) White Button does not affect the displayed character table.

The default conditions are white foreground, transparent background, not flashing, Character Table 0, normal size, not underlined, not concealed, no window/box.

SECTION 2

OPERATING MODES

2.1 INTRODUCTION

The primary operating modes available when using EUROM to handle the presentation level aspects of a videotex terminal are as follows:

(1) Stack Mode - 40 characters per row.

(2) Explicit Fill Mode - 40 characters per row.

(3) 80 Characters/Row Mode.

Modes are selected by Register R4 (see Register Map in Section 4.2). Bits 0 and R4 have the following effects.

If R4 bit 0 = 0, 40 columns per row are used.

If R4 bit 0 = 1, 80 columns per row are used.

If R4 bit 1 = 0, page data is interpreted as being Stack coded. 80 bytes (40 words) of page data are fetched from external RAM per display row during one 24 µs line flyback period.

If R4 bit 1 = 1, page data is interpreted as being 'explicit' - 160 bytes (80 words) of page data are fetched from external RAM per display row during two 24 µs line flyback periods, the page data being transferred directly to the Row Buffer without modification.

Full Field DRCS is not a primary operating mode; however, it may be used in conjunction with any of the three above.

The principles involved in the various operating modes are expanded in the following subsections.

2.2 STACK MODE (40 CHARACTERS/ROW)

To meet the videotex specification, EUROM has to display 25 rows each of 40 characters. Since 32 bits per character location are necessary (section 1.3) this would normally imply the need for 4 kbytes of external memory. This is four times the capacity of a basic ('Level 1') alphamosaic decoder memory, excluding any DRCS requirements.

Stack Architecture exploits the natural redundancy of normal text by allocating memory dynamically and (excluding DRCS) allows the external memory to be reduced to 2 Kbytes per screen. This has beneficial side effects; for example, it reduces the memory bandwidth for a given display, reducing the memory speed required or alternatively increasing the time available for microprocessor operations.

Stack Coding

In the stack coding system used in EUROM, the page memory is divided into character and attribute sections, each organised as 40-byte groups. The 40 character bytes and 40 attribute bytes together make up one display row, as shown in Figure 2.1.

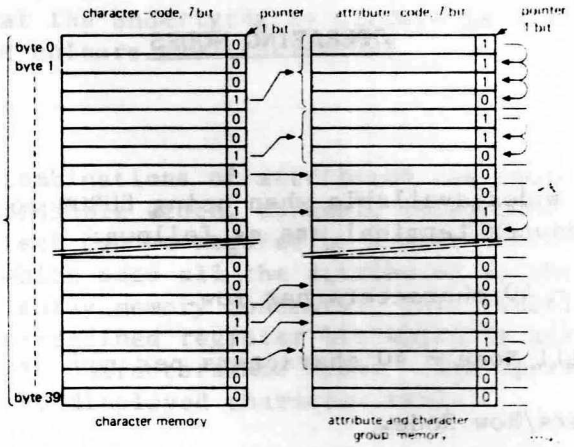


Figure 2.1 Example of dynamic allocation of memory using stack coding

Each 8-bit byte includes a pointer bit. When the pointer bit of a character byte is set, it indicates the presence of one or more attributes set at the same character position. When the pointer bit of an attribute byte is set, it indicates that there are further attributes in that group. At the beginning of a row, default attributes are set, which are then updated by the attribute bytes fetched from the stack.

An example of dynamic allocation of memory is given in Figure 2.1. The first three characters of the row have clear pointer bits. These characters will be taken from the default group of 128 (on-chip) characters, and will be displayed white on transparent background, normal size, not underlined, etc. The fourth character in the row has its pointer bit set, and so the first (or, generally, the next) attribute byte is fetched from memory. This byte has its pointer bit set, and so the next attribute byte is also fetched, and so on.

The fourth attribute byte has a clear pointer bit indicating that it is the last in the group. The next character byte is now processed. The pointer being clear, this character is displayed with the same attributes as those set for the previous one.

The stack system records only the position in a row where attribute-changes occur, with no restriction upon how many attribute-changes apply to any one character. The restriction to 40 attribute-changes in a row has now been carefully studied, and found in practice not to be an editorial limitation; it is built in to the CEPT specification as a transmission requirement.

The actual coding of attributes, a form of Huffman coding, is shown in Table 2.1.

Table 2.1 Stack coding in EUROM

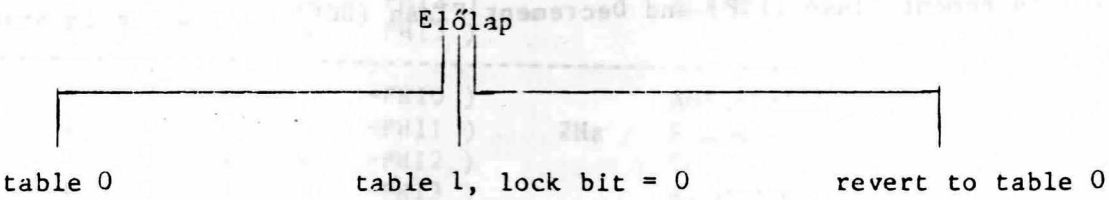
B7	B6	B5	B4	B3	B2	B1	B0	Comments
P	0	0	F4	F3	F2	F1	F0	Foreground colour Transparent = 00000
P	0	1	B4	B3	B2	B1	B0	Background colour Transparent = 00000
P	1	0	H4	H3	H2	H1	H0	FlasH
P	1	1	0	L	T2	T1	T0	Character Table and Lock bit
P	1	1	1	0	0	G	D	Size. Double HeiGht and WiDth
P	1	1	1	0	1	0	U	Underline (Lining)
P	1	1	1	0	1	1	I	Invert
P	1	1	1	1	0	0	C	Conceal
P	1	1	1	1	0	1	W	Window/Box
P	1	1	1	1	1	0	M	Marked area (not a display attribute)
P	1	1	1	1	1	1	A	Protected Area (not a display attribute)

Notes: (1) P represents Pointer bit.
(2) Binary bits 0 or 1 (in locations B1 to B6), represent identity code.
(3) Alphanumeric bits e.g. F0 to F4, represent attribute value.

Lock Bit

In Stack mode the attribute coding for character table includes a Lock Bit (see Table 2.1). The Lock Bit is used when it is necessary to switch to another character table.

If only a single character from a character table which is different from the 'pertaining' character table is required, the Lock Bit is not set and the alternative character table only applies for a single character position. The Hungarian word for 'panel' is given here as an example.



Cursor (Stack Mode)

The cursor position is defined by the Cursor Column Register (R9) and by the Cursor Row bit. The Cursor Row is identified by one of the bits (B5) in the Screen Colour byte. The Cursor Row bit is controlled by the microprocessor and so cursor on/off, cursor flash rate and mark-space ratio or even multiple cursors are all software controllable.

For normal size characters the cursor character code and attributes, specified by Registers R10 to R12 replace those at the specified character position. Any size, window/box and invert attributes are unaffected. The flash enable (FLEN) bit may be used to disable any flash effect at the cursor position.

For enlarged characters the cursor displayed depends initially on the type of equipment and then on position of the cursor, as follows.

In a simple receiving terminal application, the cursor display can occur only when the cursor position coincides with the origin of an enlarged character. Under these conditions the cursor character is displayed fully in the enlarged character area.

In editing terminals, however, in addition to the enlarged full cursor display from the character origin there is also a partial display from other positions as follows.

(a) From the bottom half of a double-height character the bottom half of the double-height cursor is displayed.

(b) From the right half of a double-width character the right side of a double-width cursor is displayed.

(c) From the top right of a double-size character, the right-hand side of a double-size cursor is displayed.

(d) From the bottom left of a double-size character, the bottom left quadrant of a double-size cursor is displayed.

(e) From the bottom right of a double-size character, the bottom right quadrant of a double-size cursor is displayed.

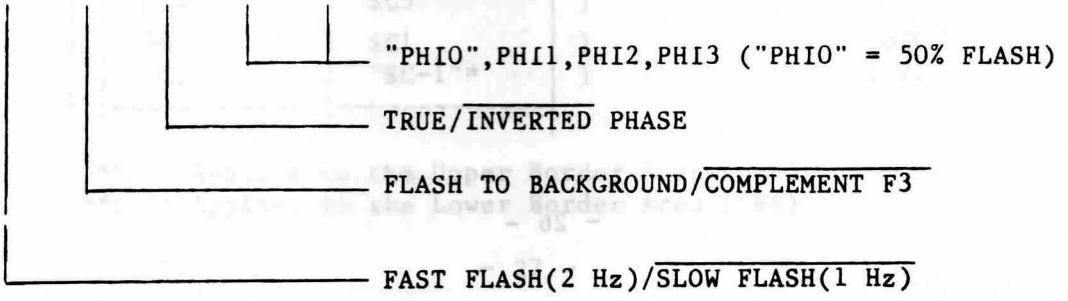
Flash Coding (Stack Mode)

When a Flash code is identified in a attribute data byte (reference Table 2.1) the five LSBs of the byte represent a complex set of Flash modes. The basic modes of flash, namely frequencies and mark-space ratios, are described in paragraph 1.11. However, for addressing purposes Table 2.2 shows the full set Flash Coding available in the Stack mode of operation.

NOTE: Increment Flash (ICF) and Decrement Flash (DCF) only apply in Stack Mode.

Table 2.2 Flash coding (Stack Mode)

H4	H3	H2	H1	H0			
0	0	0	0	0	STD	STEADY	
0	0	0	0	1	ICF	INCREMENT FLASH	1HZ
0	0	0	1	0	DCF	DECREMENT FLASH	1HZ
0	0	0	1	1		UNUSED CODE	
0	0	1	0	0	PHI0)		COLOUR
0	0	1	0	1	PHI1)	1Hz	TABLE
0	0	1	1	0	PHI2)		FLASH
0	0	1	1	1	PHI3)		
0	1	0	0	0	-PHI0)		ANTI-PHASE
0	1	0	0	1	-PHI1)	1Hz	FLASH
0	1	0	1	0	-PHI2)		TO
0	1	0	1	1	-PHI3)		BACKGROUND
0	1	1	0	0	PHI0)		FLASH
0	1	1	0	1	PHI1)	1Hz	TO
0	1	1	1	0	PHI2)		BACKGROUND
0	1	1	1	1	PHI3)		
1	0	0	0	0		STEADY	
1	0	0	0	1	ICF	INCREMENT FLASH	2HZ
1	0	0	1	0	DCF	DECREMENT FLASH	2HZ
1	0	0	1	1		UNUSED CODE	
1	0	1	0	0	PHI0)		COLOUR
1	0	1	0	1	PHI1)	2Hz	TABLE
1	0	1	1	0	PHI2)		FLASH
1	0	1	1	1	PHI3)		
1	1	0	0	0	-PHI0)		ANTI-PHASE
1	1	0	0	1	-PHI1)	2Hz	FLASH
1	1	0	1	0	-PHI2)		TO
1	1	0	1	1	-PHI3)		BACKGROUND
1	1	1	0	0	PHI0)		FLASH
1	1	1	0	1	PHI1)	2Hz	TO
1	1	1	1	0	PHI2)		BACKGROUND
1	1	1	1	1	PHI3)		



Memory Addressing (Stack Mode)

The EUROM data bus is 16 bits wide but the data which is fetched is often considered to exist in terms of bytes. The byte addressing convention is therefore of importance. The standard adopted is that of the 68000 microprocessor where the even-numbered bytes exist on the "left" or upper (most significant) part of the bus while the odd-numbered bytes are on the "right" or lower (least significant) part of the bus (see Section 5.2).

The word addresses are numerically identical to the upper byte which they contain and so there are no odd-numbered word addresses. The 17 address bits therefore provide a byte address capability of 128 Kbytes from 0 to 1FFFF, or a word address capability of 64 kwords from 0 to 1FFFE.

In Stack mode, the EUROM addresser is used for fetching chapters of data (a 'chapter' being defined as 1024 16-bit words) from external memory via the external address pins (see Table 2.3). The external memory can be considered as 64 chapters of 1024 16-bit words. In Stack mode, 1024 16-bit words are needed per page. The Display Page Chapter Pointer Register (R6) is used to define up to 64 different pages. It should be noted that address A0 is not connected to an external pin.

Table 2.3 EUROM Character/Attribute Addressing (Stack Mode)

EXTERNAL ADDRESS PINS (WORD ADDRESSES)																INTERNAL ADDRESS BIT
A16	A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
64K	32K	16K	8K	4K	2K	1K	512	256	128	64	32	16	8	4	2	1
R6 ..... bit 5					R6 bit 0											
Chapter Pointer					$\overline{\text{Char}}$/Attribute Designator											

The Chapter Pointer register R6 (see Section 4.2) provides the six Most Significant Bits (MSBs) of the address at A16 to A11.

Address bit A10 differentiates between Character and Attribute data because the first 512 x 16-bit words (1024 bytes) in the appropriate chapter of external RAM are designated as 'Character Memory' while the second 512 words are designated as 'Attribute Memory'. Screen colour data is stored as 27 of the spare bytes of memory. Table 2.4 overleaf shows the word allocations for a 1K16 memory area used in Stack Mode.

Table 2.4 Stack Mode word addressing

B15		B0	WORD ADDRESS	
	COL 0, ROW0	COL 1, ROW0	DISPLAY CHAPTER	+0
	COL 2, "	COL 3 "	" "	+2
	" "	" "	. .	.
	" "	" "	. .	.
	COL 38 "	COL 39 "	. .	+38
	ROW 1		. .	+40-78
	ROW 2		. .	+80-118
512 WORDS			. .	
			. .	
	ROW 24		. .	+960-998
"CHARACTER MEMORY"			. .	+1000
			. .	+1002
			. .	+1004
				+1006
				+1008
				+1010
		"SC25"**	)	+1012
	SC24	SC23	)	+1014
	SC22	SC21	)SCREEN	+1016
	SC20	SC19	)COLOURS	+1018
	SC18	SC17	)	+1020
	SC16	SC15	)	+1022
	ATT0, ROW0	ATT1, ROW0	DISPLAY CH. +1024	+0
	ATT2 "	ATT3 "	. .	+2
	" "	" "	. .	.
	" "	" "	. .	.
	ATT38 "	ATT39 "	. .	38
	ROW 1			40-78
	ROW 2			80-118
512 WORDS			. .	.
			. .	.
	ROW 24		. .	960-998
"ATTRIBUTE MEMORY"				1000
				1002
				1004
				1006
	SC14	SC13	)	1008
	SC12	SC11	)	1010
	SC10	SC9	)	1012
	SC8	SC7	)SCREEN	1014
	SC6	SC5	)COLOURS	1016
	SC4	SC3	)	1018
	SC2	SC1	)	1020
	SC0	"SC-1"*	)	1022

*SC-1 Applies to the Upper Border Area (UBA)
**SC25 Applies to the Lower Border Area (LBA)

Character Addresses

An individual Character Address is given by 17 address bits starting at A0. The parts of the byte address are made up as follows, with the figures in brackets indicating the number of bits used.

Character byte address location = Chapter (6) + 40 x Row (5) + Column (6).

The 25 row values are from 0 to 24 and the 40 column values are from 0 to 39.

The 40 columns x 25 rows require 1000 bytes at word addresses 0 to 998 inclusive.

Attribute Addresses

The attribute memory is similar to the character memory but starts at word address 1024, that is, address bit A10 is set.

Attribute byte address location = Chapter (6) + 1024 (A10) + 40 x Row (5) + Attribute index (6).

The Attribute Index has values 0 to 39.

Screen Colour Addresses

The 27 memory locations allocated for screen colours are split between the upper and lower halves of 2K8 memory. Screen colours for the Lower Border Area LBA and rows 15 to 24 are stored at word addresses 1012 to 1022 while screen colours for the Upper Border Area UBA and rows 0 to 14 are stored at word addresses 2032 to 2046. Thus to address a screen colour an artificial numbering system is used (see Table 2.5).

Table 2.5 Screen colour addressing (Stack Mode)

Screen Area		Artificial Number (AN)
UBA (Upper Border Area)	(SC-1)	0
Row 0	(SC 0)	1
Row 1	(SC 1)	2
:	:	:
Row 14	(SC14)	15
Row 15	(SC15)	16
:	:	:
:	:	:
Row 24	(SC24)	25
LBA (Lower Border Area)	(SC25)	26

A16 ..... A11	A10	A9 ..... A4	A3	A2	A1	A0
R6 bit 5 .... R6 bit 0 (Chapter Pointer)	$\overline{AN_4}$	1 1	$\overline{AN_3}$	$\overline{AN_2}$	$\overline{AN_1}$	$\overline{AN_0}$

By allocating the MSB of the artificial number as $\overline{A10}$ we can effectively select the upper half of memory for LBA and rows 15 to 24 and the lower half of memory for UBA and rows 0 to 14.

2.3 EXPLICIT FILL MODE (40 CHARACTERS/ROW)

This 'Direct Fill' mode can be used for 40 characters/row operation. EUROM effectively becomes a Parallel architecture machine without the inherent Stack architecture limitation of a maximum of 40 attribute changes per row. This mode is controlled by the Display Command Register (R4) (bits 0 and 1) and requires a larger amount of external RAM, that is 4 Kbytes per page, or 6 Kbytes with DRCS RAM allocated.

In this mode, the Row Buffer is arranged to be 40 x 32-bit words. This is the same as in Stack mode, but the data from external memory is transferred directly into the Row Buffer. There is no internal processing during Row Buffer filling.

The cursor in this mode must be implemented from software as is explained below. Row reveal is also ineffective in this mode.

No checking of enlarged characters is executed. If Explicit Fill mode is used for implementing CEPT Videotex, therefore, the rules relating to character size attributes (Section 1.11) must be implemented in software for enlarged characters.

Cursor (Explicit Fill Mode, 40 Characters/Row)

The cursor registers are ineffective in Explicit Fill mode (see Section 4.12) and so any cursor can not be implemented via Registers 10, 11 or 12 and Bits B5 of the Screen Colour. The cursor can be implemented from the external memory by software.

Flash Coding (Explicit Fill Mode)

This coding is similar to Stack mode except that Increment Flash (ICF) and Decrement Flash (DCF) are not implemented. The codes ICF and DCF are replaced with codes for anti-phase colour table flash (1 Hz and 2 Hz). See Table 2.6 for details.

Memory Addressing (Explicit Fill Mode)

In Explicit Fill mode the 1000 screen locations (40 columns x 25 rows) are each allocated 4 bytes of data consisting of character and attribute bits shown in Table 2.7.

Table 2.6 Flash coding Explicit Fill mode

H4	H3	H2	H1	H0			
0	0	0	0	0	STD	STEADY	
0	0	0	0	1	PHI1)		ANTI-PHASE
0	0	0	1	0	PHI2)	1Hz	COLOUR
0	0	0	1	1	PHI3)		TABLE
							FLASH
0	0	1	0	0	PHI0)		COLOUR
0	0	1	0	1	PHI1)	1Hz	TABLE
0	0	1	1	0	PHI2)		FLASH
0	0	1	1	1	PHI3)		
0	1	0	0	0	-PHI0)		ANTI-PHASE
0	1	0	0	1	-PHI1)	1Hz	FLASH
0	1	0	1	0	-PHI2)		TO
0	1	0	1	1	-PHI3)		BACKGROUND
0	1	1	0	0	PHI0)		FLASH
0	1	1	0	1	PHI1)	1Hz	TO
0	1	1	1	0	PHI2)		BACKGROUND
0	1	1	1	1	PHI3)		
1	0	0	0	0		STEADY	
1	0	0	0	1	PHI1		ANTI-PHASE
1	0	0	1	0	PHI2	2Hz	COLOUR
1	0	0	1	1	PHI3		TABLE
							FLASH
1	0	1	0	0	PHI0)		COLOUR
1	0	1	0	1	PHI1)	2Hz	TABLE
1	0	1	1	0	PHI2)		FLASH
1	0	1	1	1	PHI3)		
1	1	0	0	0	-PHI0)		ANTI-PHASE
1	1	0	0	1	-PHI1)	2Hz	FLASH
1	1	0	1	0	-PHI2)		TO
1	1	0	1	1	-PHI3)		BACKGROUND
1	1	1	0	0	PHI0)		FLASH
1	1	1	0	1	PHI1)	2Hz	TO
1	1	1	1	0	PHI2)		BACKGROUND
1	1	1	1	1	PHI3)		

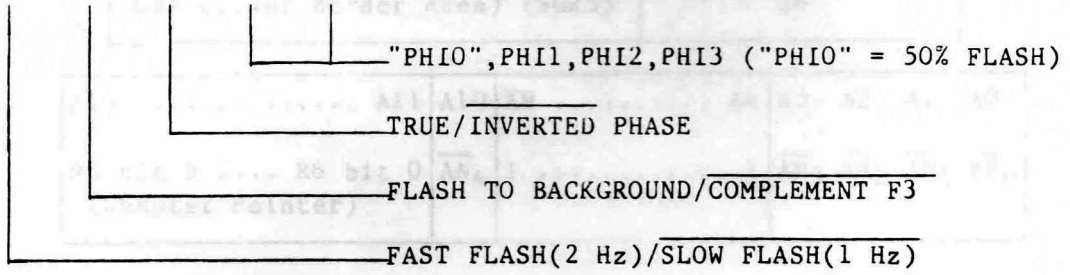


Table 2.7 Explicit Fill mode memory contents

FUNCTION	NAME	BIT
BACKGROUND COLOUR	(B4	31
	(B3	30
	(B2	29
	(B1	28
	(B0	27
FLASH	(H4	26
	(H3	25
	(H2	24
	(H1	23
	(H0	22
INVERT	I	21
FOREGROUND	(F4	20
	(F3	19
	(F2	18
	(F1	17
	(F0	16
WINDOW OR BOX	W	15
CONCEAL	C	14
UNDERLINE	U	13
TOP/BTM	T/B	12
HEIGHT	G	11
CHAR. TABLE (T2 = DRCS)	(T2	10
	(T1	9
	(T0	8
WIDTH	D	7
CHARACTER CODE	(P6	6
	(P5	5
	(P4	4
	(P3	3
	(P2	2
	(P1	1
	(P0	0

With 4 Kbytes per page, address bits A0 to A11 are needed to address the full 4 Kbytes in external memory. (See Table 2.8.)

Table 2.8 Explicit Fill addressing

EXTERNAL ADDRESS PINS (WORD ADDRESSES)															INTERNAL ADDRESS BIT
A16	A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
64K	32K	16K	8K	4K	2K	1K	512	256	128	64	32	16	8	4	2	1
R6 ..... R6 bit 5 bit 1 Chapter Pointer																

The Chapter Pointer Register (R6) provides the five Most Significant Bits (MSBs) of the address, and R6/0 is not used in this mode. An individual character or attribute byte address is defined as follows, with the figures in brackets indicating the number of bits used.

Address location = Chapter (5) + [160 x Row] + [4 x Column] (10) + byte designator (2).

The 25 row values are from 0 to 24 and the 40 column values are from 0 to 39.

The 32 bits representing a display character cell are stored as 4 successive bytes in memory as follows (and see Table 2.7 for details):

	(A1	A0		
Byte	(0	0	Bits 0-7	- Character code + width
Designator	(0	1	Bits 8-15	- Attribute 0
	(1	0	Bits 16-23	- Attribute 1
	(1	1	Bits 24-31	- Attribute 2

Screen colour data is again stored as 27 of the spare bytes of memory (see Section 2.1, Table 2.3). Table 2.6 shows the word allocations for a 4K memory page area used in Explicit Fill mode.

Table 2.8 Explicit Fill mode word addressing

B15		B0		WORD ADDRESS
	ROW 0			
AT0,	COL 0	CH, COL 0		DISPLAY CHAPTER +0
AT2,	"	AT1, "		" " +2
AT0,	COL 1	CH, COL 1		" " +4
AT2,	"	AT1, "		" " +6
.	.	. .		. . .
.	.	. .		. . .
AT0,	COL 39	CH, COL 39		" " +156
AT2,	"	AT1, "		" " +158
	ROW 1			. . +160-318
	ROW 2			. . +320-478
				. .
				. .
	ROW 24			. . +3840-3998
				. . +4000
				. . +4002
				. . +4004
				. .
				. .
				+4066
	"SC25"**		)	+4068
SC24	SC23		)	+4070
SC22	SC21		)SCREEN	+4072
SC20	SC19		)COLOURS	+4074
.	.		)	
.	.		)	
SC4	SC3		)	+4090
SC2	SC1		)	+4092
SC0	"SC-1"*		)	+4094

*SC-1 Applies to the Upper Border Area
**SC25 Applies to the Lower Border Area

2.4 80 CHARACTERS/ROW MODE

The 80 Characters/Row mode is also an Explicit Fill mode without stack coding. No additional circuitry is required, apart from a minimum of 4 Kbytes per page of external memory.

The Row Buffer is effectively rearranged to be 80 x 16-bit words. There is no data processing during row buffer filling and data from external memory is transferred directly to the row buffer. As 160 bytes are needed to fill the row buffer, two 24 µs line flyback periods per display row are required.

The cursor, as in Explicit Fill mode (Section 2.3) must be implemented by software.

In this mode, dot data is fetched from external memory, thus avoiding the need to cycle the on-chip character generator ROM any faster than 1 µs. The pixel frequency remains at 12 MHz and all characters are displayed with a 6 x 10 dot pattern. Both one bit/dot mode and two bit/mode are available.

When the 1 bit/dot mode is used, the external dot memory need only be 8 bits wide and so in the 10 lines per row mode a maximum of 204 different character matrices can be stored in a 2K8 memory (either RAM or ROM).

Cursor (80 Characters/Row Mode)

The cursor registers in 80 Characters/Row Mode are ineffective (see Section 4.12) and so any cursor must implemented in the external memory by software.

Character/Attribute Memory Addressing (80 Characters/Row Mode)

In 80 Characters/Row mode, each character cell is allocated 2 bytes of data consisting of character and attribute bits as shown in Table 2.9.

Table 2.9 80 Characters/Row mode memory contents

FUNCTION	NAME	BIT
BACKGROUND COLOUR	(B2	15
	(B1	14
	(B0	13
FLASH	(H'	12*
FOREGROUND COLOUR	(F2	11
	(F1	10
	(F0	9
UNDERLINE	(U	8
CHARACTER CODE	(P7	7
	(P6	6
	(P5	5
	(P4	4
	(P3	3
	(P2	2
	(P1	1
	(P0	0

*This bit invokes a 2 Hz colour table flash, PHI1 (code 10101).

As with the Explicit Fill mode described earlier, 4 Kbytes per page requires address bits A0 to A11 to address the full 4 Kbytes of external memory (see Table 2.10).

Table 2.10 80 Characters/Row addressing

EXTERNAL ADDRESS PINS (WORDS)																INTERNAL ADDRESS BIT
A16	A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
64K	32K	16K	8K	4K	2K	1K	512	256	128	64	32	16	8	4	2	1
R6 ..... R6 bit 5 bit 1 Chapter Pointer																

The Chapter Pointer Register (R6) provides the five Most Significant Bits (MSBs), and bit 0 is not used in this mode.

An individual character or attribute byte address is defined as follows (Note: figures in brackets depict the number of address bits).

Address location = Chapter (5) + ([160 x Row] + [2 x Column]) (11) + byte designator (1).

The 16 bits representing a display character cell are stored as 2 successive bytes in memory as follows.

```
          A0
Byte      ( 0      Bits 0-7  - Character code
Designator( 1      Bits 8-15 - Attribute
```

Screen colour data is stored as 27 of the spare bytes of memory. Table 2.11 shows the word allocations per page area used in 80 Characters/Row mode.

Table 2.11 80 Characters/Row mode word addressing

B15				B0	WORD ADDRESS
		ROW 0			
AT.	COL 0	CH.	COL 0		DISPLAY CHAPTER +0
AT.	COL 1	CH.	COL 1		" " +2
AT.	COL 2	CH.	COL 2		" " +4
AT.	COL 3	CH.	COL 3		" " +6
.	.	.	.		. . .
.	.	.	.		. . .
AT.	COL 78	CH.	COL 78		" " +156
AT.	COL 79	AT.	COL 79		" " +158
		ROW 1			. . +160-318
		ROW 2			. . +320-478
		.			. .
		.			. .
		ROW 24			. . +3840-3998
					. . +4000
					. . +4002
					. . +4004
					. .
					. .
					+4066
		"SC25"**		)	+4068
SC24		SC23		)	+4070
SC22		SC21		)SCREEN	+4072
SC20		SC19		)COLOURS	+4074
.		.		)	
.		.		)	
SC4		SC3		)	+4090
SC2		SC1		)	+4092
SC0		"SC-1"*		)	+4094

DISPLAY CHARACTER | 0 (4k) 128K BYTE OR WORD ADDRESS

*SC-1 Applies to the Upper Border Area
**SC25 Applies to the Lower Border Area

For maximum flexibility of display, the foreground and background colours are applied to different areas of the Colour Map. The Foreground uses locations 16-23 and Background colour uses locations 0-7 i.e. Background can include transparent. (See Register Map, R14, for details.)

The Flash mode incorporates colour table flash (code 10101 is used - see Table 2.5) i.e. when flash is active the foreground colour is taken from locations 24-31 of the colour look-up table.

The Underline attribute is effective in the one bit/dot mode and causes an underlining on the 10th line when operating with 10 lines per row, and on the 11th line in 12 lines per row. The underline has no effect in the 2 bit/dot mode.

Window/Box Enable (80 Characters/Row Mode)

In the 80 characters/row display mode, the Window/Box attribute (WBEN) is permanently set, but its effect may be negated by R3/5 - Window/Box Enable (or, of course, by R4/3 - White Button).

The character background colour bits are taken from locations 0 - 7 of the Colour Map, so with R3/5 = 0, setting background colour = 000 simply implies transparency to the underlying screen colour.

With R3/5 = 1, the whole Defined Display Area (DDA) becomes a window area; i.e. whenever background colour = 000, EUROM sets VDS = 0 (VDS is 'low for tv') and uses the contents of the Colour Map location 0 to define the RGB output levels. In those applications where the underlying video is absent, the visual effect of background colour = 000 is simply black.

2.5 DOT DATA ADDRESSING

Individual screen dots can be addressed by using the DRCS mode of operation. This is described in Section 3 (see subsection 3.5).

SECTION 3

DYNAMICALLY REDEFINABLE CHARACTER SETS (DRCS)

3.1 INTRODUCTION

In both 40 characters/row modes (Stack or Explicit Fill), the use of DRCS with EUROM is based on the following:

(1) 12 dot (pixel) horizontal resolution

(2) Data is stored in 1K16 'chapters' (1024 16-bit words). Normally EUROM supports two chapters of DRCS dot data, pointed to by chapter pointer registers R7 (DRCS 'A') and R8 (DRCS 'B'). More DRCS memory may be accessed using Full Field DRCS (see Section 3.7).

(3) All 8 DRCS modes may be used within a given page.

(4) DRCS characters may contain two, four or sixteen colours with 1, 2 or 4 bits/pixel.

In addition, EUROM uses relatively low-speed RAM to allow it to be used in inexpensive basic terminals as well as in more sophisticated designs.

3.2 DRCS CHARACTER CELLS

When operating with 10 lines per row the following DRCS modes are available:

Table 3.1 10 lines per row modes

Dot composition	Bits/pixel	Maximum no. of characters in 1024 16-bit words (i.e. in one 'chapter')
12 x 10	1	102
12 x 10	2	51
6 x 10	1	2 x 102
6 x 10	2	102
6 x 10	4	51
6 x 5	2	2 x 102
6 x 5	4	102

EUROM also supports an alternative 6x10 4 bits/pixel mode with a memory bit organisation more suited to a 'bit map' implementation.

When operating with 12 lines per row, fewer characters are available in each chapter of DRCS memory: see Table 3.2.

Table 3.2 12 lines per row modes

Dot composition	Bits/pixel	Maximum no. of characters in 1024 16-bit words (i.e. in one 'chapter')
12 x 12	1	85
12 x 12	2	42
6 x 12	1	2 x 85
6 x 12	2	85
6 x 12	4	42
6 x 6	2	2 x 85
6 x 6	4	85

There is also an alternative 6x12 4 bits/pixel mode.

All attributes apply to DRCS in the same manner as to normal characters but for multi-colour DRCS (i.e. the modes with 2 or 4 bits per pixel), the following additional rules apply:

(1) The whole character cell is treated as foreground colour.

(2) When the Conceal (or Flash, Invert) attribute is used, the background colour that would otherwise be pertaining is displayed.

(3) The Underline attribute has no effect. The 12x10x1 and 6x10x1 (or 12x12x1 and 6x12x1 for 12 lines/row) display modes are underlined as normal alphanumeric characters.

When operating with 2 bits/pixel, the Colour Map is indirectly addressed via DCLUT (DRCS Colour Look-up Table).

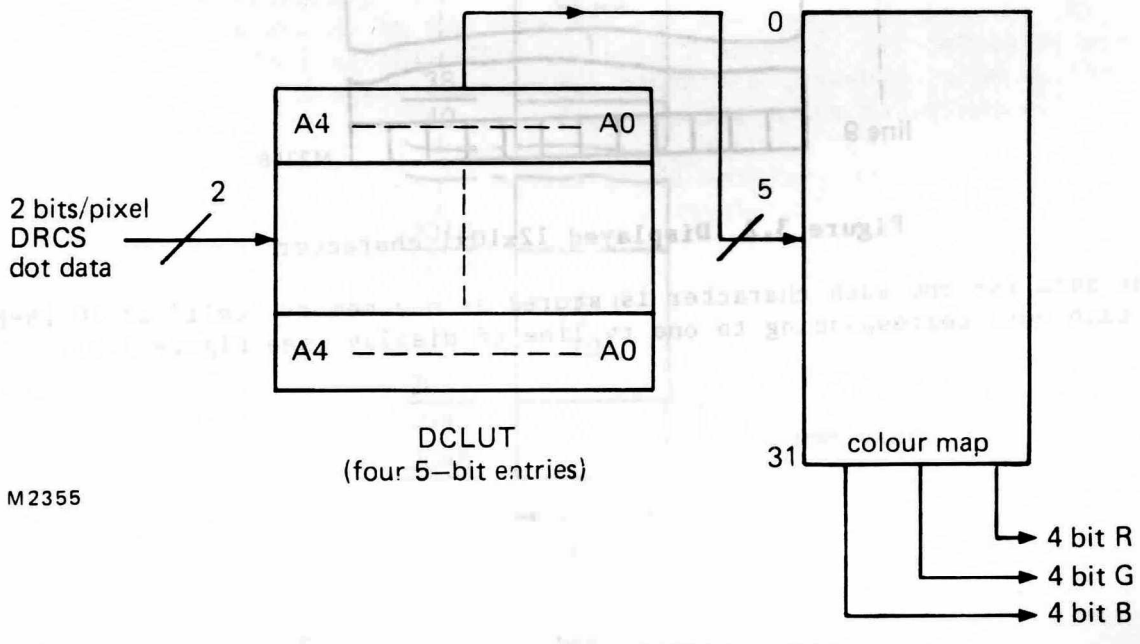


Figure 3.1 Use of DCLUT

When 4 bits/pixel colour DRCS is used, the most significant bit (A4) of the Colour Map address is taken from Register bit R3/2. Thus the colour information can be taken either from locations 0-15 or from locations 16-31 of the Colour Map. (Reference 1 requires that a BTX terminal uses location 16-31).

3.3 INTERPRETATION OF DATA BITS

When DRCS data is being accessed, it is convenient to consider each of the three Table bits T2, T1 and T0 to have an individual interpretation:

T2 = $\overline{\text{ROM}}$/DRCS. Tables 0 to 3 are the ROM-based character tables (Figures 1.4 and 1.5) and Tables 4 to 7 are the DRCS character tables.

T1 = $\overline{\text{DRCS 'A'}}$/DRCS 'B'. If T1 = 0, register R7 (DRCS 'A') is used as the DRCS chapter pointer register, and if T1 = 1, R8 (DRCS 'B') is used.

T0 = $\overline{\text{Left}}$/Right (or $\overline{\text{Even}}$/Odd). Two of the DRCS display modes (6x10x1 and 6x5x2, or the corresponding modes in 12 lines/row mode) require only half a DRCS memory character cell for storage. The memory cell is split into two with a vertical dividing line, so if one of these display modes is used, the least significant bit of the Table attribute determines which half of the memory should be used for display (left or right half, that is, even or odd bytes).

3.4 DRCS ADDRESSING (40 CHARACTERS/ROW)

The concepts of DRCS addressing are best explained by reference to the basic DRCS display mode 12x10x1 indicated in Figure 3.2.

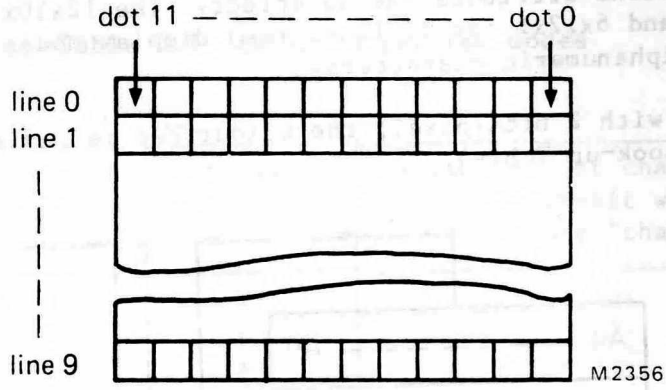


Figure 3.2 Displayed 12x10x1 character

The data for one such character is stored as one memory 'cell' of 10 16-bit words, each word corresponding to one tv line of display (see Figure 3.3).

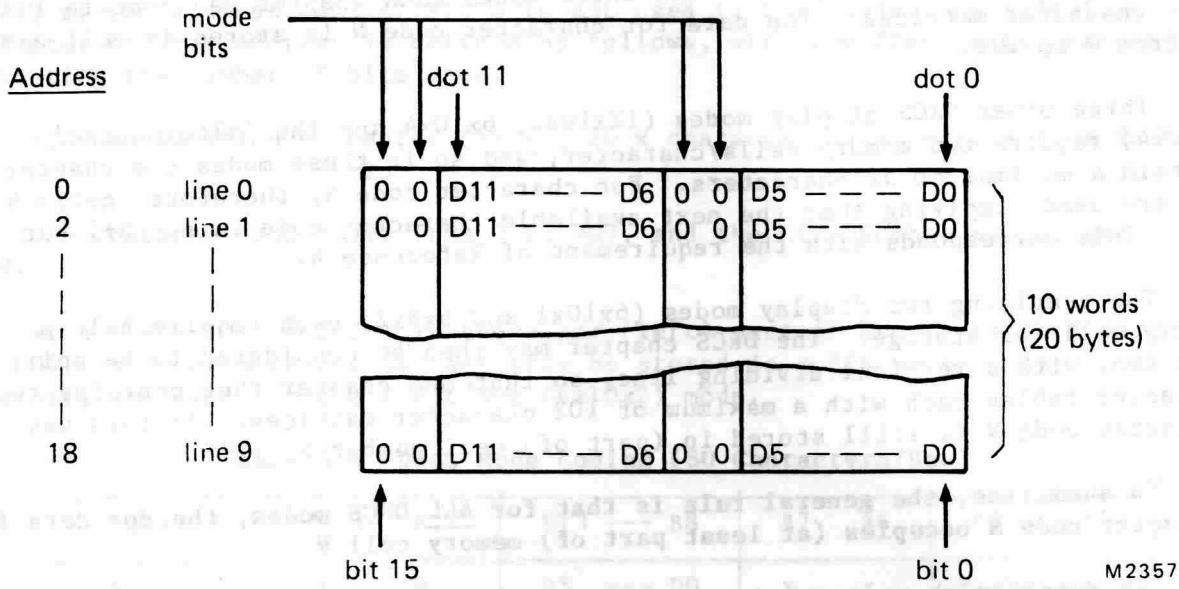


Figure 3.3 12x10x1 character corresponding to one memory cell

A DRCS chapter (see Fig.3.4) consists of 1024 16-bit words; one chapter can therefore hold 102 whole memory cells, each of 10 words (or 20 bytes).

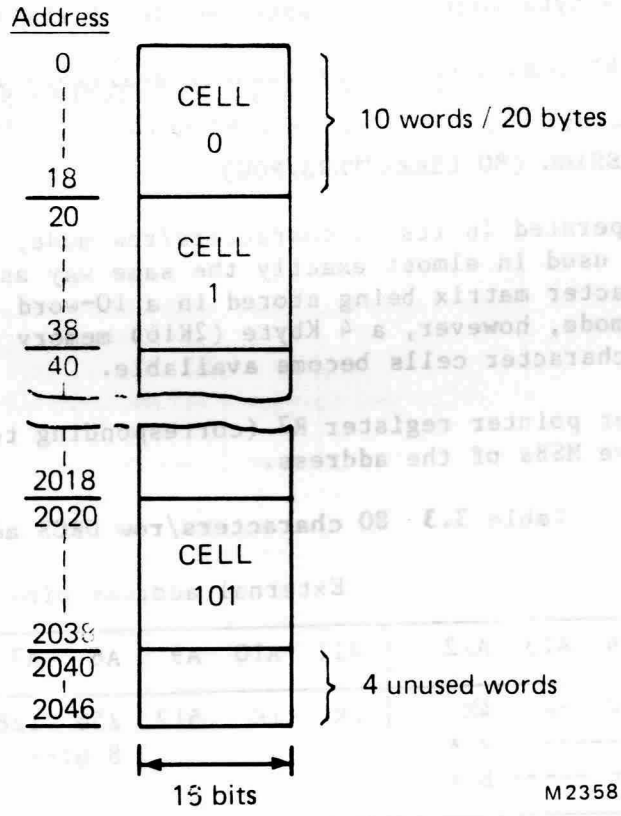


Figure 3.4 One DRCS memory chapter

Three of the DRCS display modes (12x10x1, 6x10x2 and 6x5x4) require exactly one memory cell/character, and so in these modes each chapter can contain 102 such character matrices. The data for character code N is stored in cell number N, from 0 to 101.

Three other DRCS display modes (12x10x2, 6x10x4 and the 'alternative' 6x10x4) require two memory cells/character, and so in these modes one chapter can contain a maximum of 51 characters. For character code N, therefore, cells N and N+1 are used, implying that the next available character code is not N+1 but N+2. This corresponds with the requirement of Reference 4.

The remaining two display modes (6x10x1 and 6x5x2) each require half a memory cell for storage. The DRCS chapter may then be considered to be split into two, with a vertical dividing line, so that one chapter then contains two character tables each with a maximum of 102 character matrices. In this way character code N is still stored in (part of) cell number N.

To summarise, the general rule is that for ALL DRCS modes, the dot data for character code N occupies (at least part of) memory cell N.

The data for character N is stored in cell N. For 10 lines/row each cell is 20 bytes. With line-number values L from 0 to 9, the byte address of character N is given as follows.

Character N byte address = Chapter offset + 20xN + 2xL

For 12 lines/row each cell is 12 bytes. With line-number values L from 0 to 11, the byte address of character N is given as follows.

Character N byte address = Chapter offset + 24N +2L

There are 85 memory cells per chapter available when using the 12 lines per row mode.

3.5 DRCS ADDRESSING (80 CHARACTERS/ROW)

When EUROM is operated in its 80 characters/row mode, the external dot memory (ROM or RAM) is used in almost exactly the same way as in the 40 characters/row mode, each character matrix being stored in a 10-word 'cell'. In the 80 characters/row mode, however, a 4 Kbyte (2K16) memory space is used so that at 10 lines/row, 204 character cells become available.

DRCS chapter pointer register R7 (corresponding to DRCS chapter 'A') provides the five MSBs of the address.

Table 3.3 80 characters/row DRCS addressing

External address pins

A16	A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1
64K	32K	16K	8K	4K	2K	1K	512	256	128	64	32	16	8	4	2
R7 ---------- R7*								8 bits					4 bits with A0		
Bit 6 ---------- Bit 1															

The Character Pointer Register 7 provides the five Most Significant Bits (MSBs) of the address, and R7 bit 0 is not used in this mode. An individual character or byte address is defined as follows, with the figures in brackets indicating the number of bits used.

Address location = Chapter (5) + ([20 x character code] + [2 x line Number])
(12)

The character codes are from 0 to 203, and the Line Number values are from 0 to 9.

With the 6x10x1 mode, only the upper (left) part of memory is used so that the 204 character matrices may actually be stored in a 2K8 memory space. All 16 data bits are used for the 2 bit/dot (6x10x2) mode.

Table 3.4 DRCS Mode Coding (80 Characters/Row)

Mode	B15	B14	B13 --- B8	B7	B6	B5 --- B0
1 bit/dot 6x10x1	1	0	D5 --- D0	x	x	x x x x x x
2 bits/dot 6x10x2	1	1	D5 --- D0 DCLUT A0	x	x	D5 --- D0 DCLUT A1

When 12 lines/row is used, 170 6x12 character matrices can be stored in a 2K memory space.

3.6 DRCS MODE CODING

A summary of DRCS Mode coding examples is found in Figure 3.5. A more detailed explanation in the form of memory coding tables, is found in Figures 3.8 to 3.24 in the following pages.

Mode	B15	B14	B13 ..... B8	B7	B6	B5 ..... B0	Comments
12x10x1	0	0	D11 ..... D6	0	0	D5 ..... D0	One memory cell
							Two memory cells
12x10x2	0	0	D11 ..... D6	0	1	D5 ..... D0	1st cell defines D-CLUT DA0
							2nd cell defines D-CLUT DA1
							One memory cell
Even Byte 6x5x2 / Odd Byte 6x5x2	1	1	D5 ..... D0 Table 4 or 6 D-CLUT DA0	1	1	D5 ..... D0 Table 5 or 7 D-CLUT DA0	1st, 3rd, 5th, 7th and 9th words of cell define D-CLUT DA0
			D5 ..... D0 Table 4 or 6 D-CLUT DA1			D5 ..... D0 Table 5 or 7 D-CLUT DA1	2nd, 4th, 6th, 8th and 10th words of cell define D-CLUT DA1
Even Byte 6x5x2 / Odd Byte 6x10x1	1	1	D5 ..... D0 Table 4 or 6 D-CLUT DA0	1	0	D5 ..... D0 Table 5 or 7	One memory cell Two character matrices. 6x5x2 data stored as above
			D5 .... D0 Table 4 or 6 D-CLUT DA1				
Even byte 6x10x1 / Odd byte 6x5x2	1	0	D5 .... D0 Table 4 or 6	1	1	D5 ..... D0 Table 5 or 7 D-CLUT DA0	One memory cell Two character matrices.
						D5 ..... D0 Table 5 or 7 D-CLUT DA1	
Even byte 6x10x1 / Odd byte 6x10x1	1	0	D5 .... D0 Table 4 or 6	1	0	D5 ..... D0 Table 5 or 7	One memory cell Two character matrices.

Figure 3.5A DRCS mode coding (continued overleaf)

Mode	B15	B14	B13 ..... B8	B7	B6	B5 ..... B0	Comments
							One memory cell
6x10x2	1	0	D5 ..... D0 D-CLUT DA0	0	1	D5 ..... D0 D-CLUT DA1	Even bytes define D-CLUT DA0 Odd bytes define D-CLUT DA1
							Two memory cells
6x10x4	0	0	D5 .... D0 C'LR.MAP A0	1	1	D5 ..... D0 C'LR.MAP A1	First cell defines colour map A0 and A1
			D5 .... D0 C'LR.MAP A2			D5 ..... D0 C'LR.MAP A3	Second cell defines colour map A2 and A3
							One memory cell
6x5x4	1	1	D5 .... D0 C'LR.MAP A0	0	1	D5 ..... D0 C'LR.MAP A1	1st, 3rd, 5th, 7th and 9th words define colour map A0 and A1
			D5 .... D0 C'LR.MAP A2			D5 ..... D0 C'LR.MAP A3	2nd, 4th, 6th, 8th and 10th words define colour map A2 and A3
							Two memory cells
Alternative 6x10x4	0	1 xx	D5 A3 ... A0		D4 A3 ... A0	D3 A3 ... A0	First cell defines pixels D5, D4 and D3
			D2 A3 ... A0		D1 A3 ... A0	D0 A3 ... A0	Second cell defines pixels D2, D1 and D0

Figure 3.5B DRCS mode coding (continued)

Notes:

1. A memory cell is defined as 10 consecutive words (i.e. 20 Bytes) for 10 lines per Row.
2. Bn means memory bit n. (0 <= n <= 15).
3. Dn means dot data for pixel n. (0 <= n <= 5, or 0 <= n <= 11).
4. An means Colour Map address bit n. (0 <= n <= 4).
5. DAn means D-CLUT address bit n. (0 <= n <= 1).
6. C'LR.MAP means Colour Map.
7. When the 6x5x4 or the 6x10x4 DRCS modes are used, the colour map address A4 is taken from register R3/1.

Mode	B15 B14	B13 ..... B8	B7 B6	B5 ..... B0	Comments
12x10x1	0 0	D11 ..... D6	0 0	D5 ..... D0	One memory cell

LINE NO.	DISPLAY	MEMORY DATA MODE MODE
0	100000000011	00 100000 00 000011
1	010000000110	00 010000 00 000110
2	001000001100	00 001000 00 001100
3	000100011000	00 000100 00 011000
4	000010110000	00 000010 00 110000
5	000001100000	00 000001 00 100000
6	000011100000	00 000011 00 100000
7	000110010000	00 000110 00 010000
8	001100001000	00 001100 00 001000
9	011000000100	00 011000 00 000100

12x10x1 MODE 10 Lines Per Row

Mode	B15 B14	B13 ..... B8	B7 B6	B5 ..... B0	Comments
12x12x1	0 0	D11 ..... D6	0 0	D5 ..... D0	One memory cell

LINE NO.	DISPLAY	MEMORY DATA MODE MODE
0	100000000011	00 100000 00 000011
1	010000000110	00 010000 00 000110
2	001000001100	00 001000 00 001100
3	000100011000	00 000100 00 011000
4	000010110000	00 000010 00 110000
5	000001100000	00 000001 00 100000
6	000011100000	00 000011 00 100000
7	000110010000	00 000110 00 010000
8	001100001000	00 001100 00 001000
9	011000000100	00 011000 00 000100
10	110000000010	00 110000 00 000010
11	100000000001	00 100000 00 000001

12x12x1 MODE 12 Lines Per Row

Figure 3.6 DRCS memory coding example

Mode	B15 B14	B13 ..... B8	B7 B6	B5 ..... B0	Comments
					Two memory cells
12x10x2	0 0	D11 ..... D6	0 1	D5 ..... D0	1st cell defines D-CLUT DA0 2nd cell defines D-CLUT DA1

LINE NO.	DISPLAY	MEMORY DATA MODE MODE
0	200000000001	00 000000 01 000001
1	220000000011	00 000000 01 000011
2	222000000111	00 000000 01 000111
3	222200001111	00 000000 01 001111
4	222220011111	00 000000 01 011111
5	222222111111	00 000000 01 111111
6	222223311111	00 000001 01 111111
7	222233331111	00 000011 01 111111
8	222333333111	00 000111 01 111111
9	223333333311	00 001111 01 111111

CELL N

MODE MODE

00 100000 01 000000
00 110000 01 000000
00 111000 01 000000
00 111100 01 000000
00 111110 01 000000
00 111111 01 000000
00 111111 01 100000
00 111111 01 110000
00 111111 01 111000
00 111111 01 111100

CELL N+1

12x10x2 MODE 10 Lines Per Row

Figure 3.7 DRCS memory coding example

Mode	B15	B14	B13 ..... B8	B7	B6	B5 ..... B0	Comments
							Two memory cells
12x10x2	0	0	D11 ..... D6	0	1	D5 ..... D0	1st cell defines D-CLUT DA0 2nd cell defines D-CLUT DA1

LINE NO.	DISPLAY	MEMORY DATA MODE MODE
0	200000000001	00 000000 01 000001
1	220000000011	00 000000 01 000011
2	222000000111	00 000000 01 000111
3	222200001111	00 000000 01 001111
4	222220011111	00 000000 01 011111
5	222222111111	00 000000 01 111111
6	222223311111	00 000001 01 111111
7	222233331111	00 000011 01 111111
8	222333333111	00 000111 01 111111
9	223333333311	00 001111 01 111111
10	233333333331	00 011111 01 111111
11	333333333333	00 111111 01 111111

CELL N

MODE	MODE
00	100000 01 000000
00	110000 01 000000
00	111000 01 000000
00	111100 01 000000
00	111110 01 000000
00	111111 01 000000
00	111111 01 100000
00	111111 01 110000
00	111111 01 111000
00	111111 01 111100
00	111111 01 111110
00	111111 01 111111

CELL N+1

12x12x2 MODE 12 Lines Per Row

Figure 3.8 DRCS memory coding example

Mode		B15 B14	B13 ..... B8	B7 B6	B5 ..... B0	Comments
						One memory cell
Even Byte 6x5x2	Odd Byte 6x5x2	1 1	D5 ..... D0 Table 4 or 6 D-CLUT DA0	1 1	D5 ..... D0 Table 5 or 7 D-CLUT DA0	1st, 3rd, 5th, 7th and 9th words of cell define D-CLUT DA0
			D5 ..... D0 Table 4 or 6 D-CLUT DA1		D5 ..... D0 Table 5 or 7 D-CLUT DA1	2nd, 4th, 6th, 8th and 10th words of cell define D-CLUT DA1

LINE NO.	DISPLAY	MEMORY DATA MODE	MEMORY DATA MODE
0	2 0 0 0 0 1		
1			
2	2 2 0 0 1 1		
3			
4	2 2 2 1 1 1		
5			
6	2 2 3 3 1 1		
7		11 000001	11 000001
8	2 3 3 3 3 1	11 100000	11 100000
9		11 000011	11 000011
	6x5x2 (Left Half)	11 110000	11 110000
		11 000111	11 000111
		11 111000	11 111000
		11 001111	11 001111
0	2 0 0 0 0 1	11 111100	11 111100
1		11 011111	11 011111
2	2 2 0 0 1 1	11 111110	11 111110
3			
4	2 2 2 1 1 1		
5			
6	2 2 3 3 1 1		
7			
8	2 3 3 3 3 1		
9			

6x5x2 (Right Half) 10 Lines Per Row

Figure 3.9 DRCS memory coding example

Mode		B15 B14	B13 ..... B8	B7 B6	B5 ..... B0	Comments
						One memory cell
Even Byte 6x6x2	Odd Byte 6x6x2	1 1	D5 ..... D0 Table 4 or 6 D-CLUT DA0	1 1	D5 ..... D0 Table 5 or 7 D-CLUT DA0	1st, 3rd, 5th, 7th and 9th words of cell define D-CLUT DA0
			D5 ..... D0 Table 4 or 6 D-CLUT DA1		D5 ..... D0 Table 5 or 7 D-CLUT DA1	2nd, 4th, 6th, 8th and 10th words of cell define D-CLUT DA1

LINE NO.	DISPLAY	MEMORY DATA MODE	MODE
0	2 0 0 0 0 1		
1			
2	2 2 0 0 1 1		
3			
4	2 2 2 1 1 1		
5			
6	2 2 3 3 1 1		
7			
8	2 3 3 3 3 1		
9		11 000001	11 000001
10	3 3 3 3 3 3	11 100000	11 100000
11		11 000011	11 000011
		11 110000	11 110000
	6x6x2 (Left Half)	11 000111	11 000111
		11 111000	11 111000
		11 001111	11 001111
		11 111100	11 111100
0	2 0 0 0 0 1	11 011111	11 011111
1		11 111110	11 111110
2	2 2 0 0 1 1	11 111111	11 111111
3		11 111111	11 111111
4	2 2 2 1 1 1		
5			
6	2 2 3 3 1 1		
7			
8	2 3 3 3 3 1		
9			
10	3 3 3 3 3 3		
11			

6x6x2 (Right Half) 12 Lines Per Row

Figure 3.10 DRCS memory coding example

Mode		B15 B14	B13 ..... B8	B7 B6	B5 ..... B0	Comments
Even Byte	Odd Byte					One memory cell
6x5x2	6x10x1	1 1	D5 ..... D0 Table 4 or 6 D-CLUT DA0 --- D5 .... D0 Table 4 or 6 D-CLUT DA1	1 0	D5 ..... D0 Table 5 or 7	Two character matrices. 6x5x2 data stored as in Figure 3.9

LINE NO.	DISPLAY	MEMORY DATA MODE MODE
0	2 0 0 0 0 1	
1		
2	2 2 0 0 1 1	
3		
4	2 2 2 1 1 1	
5		
6	2 2 3 3 1 1	
7		11 000001 10 100011
8	2 3 3 3 3 1	11 100000 10 010110
9		11 000011 10 001100
	6x5x2 (Left Half)	11 110000 10 011100
		11 000111 10 110010
		11 111000 10 100001
		11 001111 10 100011
0	1 0 0 0 1 1	11 111100 10 010110
1	0 1 0 1 1 0	11 011111 10 001100
2	0 0 1 1 0 0	11 111110 10 011100
3	0 1 1 1 0 0	
4	1 1 0 0 1 0	
5	1 0 0 0 0 1	
6	1 0 0 0 1 1	
7	0 1 0 1 1 0	
8	0 0 1 1 0 0	
9	0 1 1 1 0 0	
	6x10x1 (Right Half) 10 Lines Per Row	

Figure 3.11 DRCS memory coding example

Mode		B15 B14	B13 ..... B8	B7 B6	B5 ..... B0	Comments
Even Byte 6x6x2	Odd Byte 6x12x1	1 1	D5 ..... D0 Table 4 or 6 D-CLUT DA0 --- D5 .... D0 Table 4 or 6 D-CLUT DA1	1 0	D5 ..... D0 Table 5 or 7	One memory cell Two character matrices. 6x6x2 data stored as in Figure 3.10

LINE NO.	DISPLAY	MEMORY DATA MODE MODE
0	2 0 0 0 0 1	
1		
2	2 2 0 0 1 1	
3		
4	2 2 2 1 1 1	
5		
6	2 2 3 3 1 1	
7		
8	2 3 3 3 3 1	
9		11 000001 10 100011
10	3 3 3 3 3 3	11 100000 10 010110
11		11 000011 10 001100
		11 110000 10 011100
	6x6x2 (Left Half)	11 000111 10 110010
		11 111000 10 100001
		11 001111 10 100011
		11 111100 10 010110
		11 011111 10 001100
0	1 0 0 0 1 1	11 111110 10 011100
1	0 1 0 1 1 0	11 111111 10 110010
2	0 0 1 1 0 0	11 111111 10 100001
3	0 1 1 1 0 0	
4	1 1 0 0 1 0	
5	1 0 0 0 0 1	
6	1 0 0 0 1 1	
7	0 1 0 1 1 0	
8	0 0 1 1 0 0	
9	0 1 1 1 0 0	
10	1 1 0 0 1 0	
11	1 0 0 0 0 1	

6x12x1 (Right Half) 12 Lines Per Row

Figure 3.12 DRCS memory coding example

Mode		B15 B14	B13 ..... B8	B7 B6	B5 ..... B0	Comments
Even byte 6x10x1	Odd byte 6x5x2	1 0	D5 .... D0 Table 4 or 6	1 1	D5 ..... D0 Table 5 or 7 D-CLUT DA0 --- D5 ..... D0 Table 5 or 7 D-CLUT DA1	One memory cell Two character matrices. 6x5x2 data stores as in Figure 3.9

LINE NO.	DISPLAY	MEMORY DATA MODE MODE
0	1 0 0 0 1 1	
1	0 1 0 1 1 0	
2	0 0 1 1 0 0	
3	0 1 1 1 0 0	
4	1 1 0 0 1 0	
5	1 0 0 0 0 1	
6	1 0 0 0 1 1	
7	0 1 0 1 1 0	10 100011 11 000001
8	0 0 1 1 0 0	10 010110 11 100000
9	0 1 1 1 0 0	10 001100 11 000011
		10 011100 11 110000
	6x10x1 (Left Half)	10 110010 11 000111
		10 100001 11 111000
		10 100011 11 001111
0	2 0 0 0 0 1	10 010110 11 111100
1		10 001100 11 011111
2	2 2 0 0 1 1	10 011100 11 111110
3		
4	2 2 2 1 1 1	
5		
6	2 2 3 3 1 1	
7		
8	2 3 3 3 3 1	
9		
	6x5x2 (Right Half)	
	10 Lines Per Row	

Figure 3.13 DRCS memory coding example

Mode		B15 B14	B13 ..... B8	B7 B6	B5 ..... B0	Comments
Even byte	Odd byte					One memory cell
6x12x1	6x6x2	1 0	D5 .... D0 Table 4 or 6	1 1	D5 ..... D0 Table 5 or 7 D-CLUT DA0 --- D5 ..... D0 Table 5 or 7 D-CLUT DA1	Two character matrices. 6x6x2 data stored as in Figure 3.10

LINE NO.	DISPLAY	MEMORY DATA MODE MODE
0	1 0 0 0 1 1	
1	0 1 0 1 1 0	
2	0 0 1 1 0 0	
3	0 1 1 1 0 0	
4	1 1 0 0 1 0	
5	1 0 0 0 0 1	
6	1 0 0 0 1 1	
7	0 1 0 1 1 0	
8	0 0 1 1 0 0	
9	0 1 1 1 0 0	10 100011 11 000001
10	1 1 0 0 1 0	10 010110 11 100000
11	1 0 0 0 0 1	10 001100 11 000011
		10 011100 11 110000
	6x12x1 (Left Half)	10 110010 11 000111
		10 100001 11 111000
		10 100011 11 001111
		10 010110 11 111100
		10 001100 11 011111
0	2 0 0 0 0 1	10 011100 11 111110
1		10 110010 11 111111
2	2 2 0 0 1 1	10 100001 11 111111
3		
4	2 2 2 1 1 1	
5		
6	2 2 3 3 1 1	
7		
8	2 3 3 3 3 1	
9		
10	3 3 3 3 3 3	
11		
	6x6x2 (Right Half)	
	12 Lines Per Row	

Figure 3.14 DRCS memory coding example

Mode		B15 B14	B13 ..... B8	B7 B6	B5 ..... B0	Comments
Even byte	Odd byte					One memory cell
6x10x1	6x10x1	1 0	D5 .... D0 Table 4 or 6	1 0	D5 ..... D0 Table 5 or 7	Two character matrices.

LINE NO.	DISPLAY	MEMORY DATA MODE	MEMORY DATA MODE
0	1 0 0 0 1 1		
1	0 1 0 1 1 0		
2	0 0 1 1 0 0		
3	0 1 1 1 0 0		
4	1 1 0 0 1 0		
5	1 0 0 0 0 1		
6	1 0 0 0 1 1		
7	0 1 0 1 1 0	10 100011	10 100011
8	0 0 1 1 0 0	10 010110	10 010110
9	0 1 1 1 0 0	10 001100	10 001100
		10 011100	10 011100
	6x10x1 (Left Half)	10 110010	10 110010
		10 100001	10 100001
		10 100011	10 100011
0	1 0 0 0 1 1	10 010110	10 010110
1	0 1 0 1 1 0	10 001100	10 001100
2	0 0 1 1 0 0	10 011100	10 011100
3	0 1 1 1 0 0		
4	1 1 0 0 1 0		
5	1 0 0 0 0 1		
6	1 0 0 0 1 1		
7	0 1 0 1 1 0		
8	0 0 1 1 0 0		
9	0 1 1 1 0 0		

6x10x1 (Right Half)

10 Lines Per Row

Figure 3.15 DRCS memory coding example

Mode	B15 B14	B13 ..... B8	B7 B6	B5 ..... B0	Comments
Even byte Odd byte					One memory cell
6x12x1 6x12x1	1 0	D5 .... D0 Table 4 or 6	1 0	D5 ..... D0 Table 5 or 7	Two character matrices.

LINE NO.	DISPLAY	MEMORY DATA MODE MODE
0	1 0 0 0 1 1	
1	0 1 0 1 1 0	
2	0 0 1 1 0 0	
3	0 1 1 1 0 0	
4	1 1 0 0 1 0	
5	1 0 0 0 0 1	
6	1 0 0 0 1 1	
7	0 1 0 1 1 0	
8	0 0 1 1 0 0	
9	0 1 1 1 0 0	10 100011 10 100011
10	1 1 0 0 1 0	10 010110 10 010110
11	1 0 0 0 0 1	10 001100 10 001100
		10 011100 10 011100
	6x12x1 (Left Half)	10 110010 10 110010
		10 100001 10 100001
		10 100011 10 100011
		10 010110 10 010110
		10 001100 10 001100
0	1 0 0 0 1 1	10 011100 10 011100
1	0 1 0 1 1 0	10 110010 10 110010
2	0 0 1 1 0 0	10 100001 10 100001
3	0 1 1 1 0 0	
4	1 1 0 0 1 0	
5	1 0 0 0 0 1	
6	1 0 0 0 1 1	
7	0 1 0 1 1 0	
8	0 0 1 1 0 0	
9	0 1 1 1 0 0	
10	1 1 0 0 1 0	
11	1 0 0 0 0 1	
	6x12x1 (Right Half)	
	12 Lines Per Row	

Figure 3.16 DRCS memory coding example

Mode	B15 B14	B13 ..... B8	B7 B6	B5 ..... B0	Comments
					One memory cell
6x10x2	1 0	D5 ..... D0 D-CLUT DA0	0 1	D5 ..... D0 D-CLUT DA1	Even bytes define D-CLUT DA0 Odd bytes define D-CLUT DA1

LINE NO.		MODE	MODE
0	1 0 0 0 0 2	10 100000	01 000001
1	1 1 0 0 2 2	10 110000	01 000011
2	1 1 1 2 2 2	10 111000	01 000111
3	1 1 3 3 2 2	10 111100	01 001111
4	1 3 3 3 3 2	10 111110	01 011111
5	3 3 3 3 3 3	10 111111	01 111111
6	1 0 0 0 0 2	10 100000	01 000001
7	1 1 0 0 2 2	10 110000	01 000011
8	1 1 1 2 2 2	10 111000	01 000111
9	1 1 3 3 2 2	10 111100	01 001111

6x10x2 MODE 10 Lines Per Row

Mode	B15 B14	B13 ..... B8	B7 B6	B5 ..... B0	Comments
					One memory cell
6x12x2	1 0	D5 ..... D0 D-CLUT DA0	0 1	D5 ..... D0 D-CLUT DA1	Even bytes define D-CLUT DA0 Odd bytes define D-CLUT DA1

LINE NO.	DISPLAY	MEMORY DATA MODE	MODE
0	1 0 0 0 0 2	10 100000	01 000001
1	1 1 0 0 2 2	10 110000	01 000011
2	1 1 1 2 2 2	10 111000	01 000111
3	1 1 3 3 2 2	10 111100	01 001111
4	1 3 3 3 3 2	10 111110	01 011111
5	3 3 3 3 3 3	10 111111	01 111111
6	1 0 0 0 0 2	10 100000	01 000001
7	1 1 0 0 2 2	10 110000	01 000011
8	1 1 1 2 2 2	10 111000	01 000111
9	1 1 3 3 2 2	10 111100	01 001111
10	1 3 3 3 3 2	10 111110	01 011111
11	3 3 3 3 3 3	10 111111	01 111111

6x12x2 MODE 12 Lines Per Row

Figure 3.17 DRCS memory coding example

Mode	B15 B14	B13 ..... B8	B7 B6	B5 ..... B0	Comments
					Two memory cells
6x10x4	0 0	D5 .... D0 C'LR.MAP A0	1 1	D5 ..... D0 C'LR.MAP A1	First cell defines colour map A0 and A1
		D5 .... D0 C'LR.MAP A2		D5 ..... D0 C'LR.MAP A3	Second cell defines colour map A2 and A3

Note C'LR.MAP = Colour Map

LINE NO.	DISPLAY	MEMORY DATA MODE MODE
0	5 4 3 2 1 0	00 101010 11 001100
1	B A 9 8 7 6	00 101010 11 110011
2	1 0 F E D C	00 101010 11 001100
3	7 6 5 4 3 2	00 101010 11 110011
4	D C B A 9 8	00 101010 11 001100
5	3 2 1 0 F E	00 101010 11 110011
6	9 8 7 6 5 4	00 101010 11 001100
7	F E D C B A	00 101010 11 110011
8	5 4 3 2 1 0	00 101010 11 001100
9	B A 9 8 7 6	00 101010 11 110011

6x10x4 MODE 10 Lines Per Row CELL N

00 110000 11 000000
00 000011 11 111100
00 001111 11 001111
00 111100 11 000000
00 110000 11 111111
00 000011 11 000011
00 001111 11 110000
00 111100 11 111111
00 110000 11 000000
00 000011 11 111100

CELL N+1

Figure 3.18 DRCS memory coding example

Mode	B15 B14	B13 ..... B8	B7 B6	B5 ..... B0	Comments
					Two memory cells
6x12x4	0 0	D5 .... D0 C'LR.MAP A0	1 1	D5 ..... D0 C'LR.MAP A1	First cell defines colour map A0 and A1
		D5 .... D0 C'LR.MAP A2		D5 ..... D0 C'LR.MAP A3	Second cell defines colour map A2 and A3

Note C'LR.MAP = Colour Map

LINE NO.	DISPLAY	MEMORY DATA MODE MODE
0	5 4 3 2 1 0	00 101010 11 001100
1	B A 9 8 7 6	00 101010 11 110011
2	1 0 F E D C	00 101010 11 001100
3	7 6 5 4 3 2	00 101010 11 110011
4	D C B A 9 8	00 101010 11 001100
5	3 2 1 0 F E	00 101010 11 110011
6	9 8 7 6 5 4	00 101010 11 001100
7	F E D C B A	00 101010 11 110011
8	5 4 3 2 1 0	00 101010 11 001100
9	B A 9 8 7 6	00 101010 11 110011
10	1 0 F E D C	00 101010 11 001100
11	7 6 5 4 3 2	00 101010 11 110011

6x12x4 MODE 12 Lines Per Row CELL N

00 110000 11 000000
00 000011 11 111100
00 001111 11 001111
00 111100 11 000000
00 110000 11 111111
00 000011 11 000011
00 001111 11 110000
00 111100 11 111111
00 110000 11 000000
00 000011 11 111100
00 001111 11 001111
00 111100 11 000000

CELL N+1

Figure 3.19 DRCS memory coding example

Mode	B15 B14	B13 ..... B8	B7 B6	B5 ..... B0	Comments
					One memory cell
6x5x4	1 1	D5 .... D0 C'LR.MAP A0	0 1	D5 ..... D0 C'LR.MAP A1	1st, 3rd, 5th, 7th and 9th words define colour map A0 and A1
		D5 .... D0 C'LR.MAP A2		D5 ..... D0 C'LR.MAP A3	2nd, 4th, 6th, 8th and 10th words define colour map A2 and A3

Note C'LR.MAP = Colour Map

LINE NO.	DISPLAY	MEMORY DATA MODE MODE
0	5 4 3 2 1 0	11 101010 01 001100
1		11 110000 01 000000
2	B A 9 8 7 6	11 101010 01 110011
3		11 000011 01 111100
4	1 0 F E D C	11 101010 01 001100
5		11 001111 01 001111
6	7 6 5 4 3 2	11 101010 01 110011
7		11 111100 01 000000
8	D C B A 9 8	11 101010 01 001100
9		11 110000 01 111111

6x5x4 MODE 10 Lines Per Row

Figure 3.20A DRCS memory coding examples

Mode	B15 B14	B13 ..... B8	B7 B6	B5 ..... B0	Comments
					One memory cell
6x6x4	1 1	D5 .... D0 C'LR.MAP A0	0 1	D5 ..... D0 C'LR.MAP A1	1st, 3rd, 5th, 7th and 9th words define colour map A0 and A1
		D5 .... D0 C'LR.MAP A2		D5 ..... D0 C'LR.MAP A3	2nd, 4th, 6th, 8th and 10th words define colour map A2 and A3

Note C'LR.MAP = Colour Map

LINE NO.	DISPLAY	MEMORY DATA MODE MODE
0	5 4 3 2 1 0	11 101010 01 001100
1		11 110000 01 000000
2	B A 9 8 7 6	11 101010 01 110011
3		11 000011 01 111100
4	1 0 F E D C	11 101010 01 001100
5		11 001111 01 001111
6	7 6 5 4 3 2	11 101010 01 110011
7		11 111100 01 000000
8	D C B A 9 8	11 101010 01 001100
9		11 110000 01 111111
10	3 2 1 0 F E	11 101010 01 110011
11		11 000011 01 000011

6x6x4 MODE 12 Lines Per Row

Figure 3.20B DRCS memory coding example

Mode	B15 B12	B11 .. B8	B7 ... B4	B3 ... B0	Comments
					Two memory cells
Alternative 6x10x4	0 1 xx	D5 A3 ... A0	D4 A3 ... A0	D3 A3 ... A0	First cell defines pixels D5, D4 and D3
		D2 A3 ... A0	D1 A3 ... A0	D0 A3 ... A0	Second cell defines pixels D2, D1 and D0

LINE NO.	DISPLAY	MEMORY DATA MODE
0	5 4 3 2 1 0	01XX 010101000011
1	B A 9 8 7 6	01XX 101110101001
2	1 0 F E D C	01XX 000100001111
3	7 6 5 4 3 2	01XX 011101100101
4	D C B A 9 8	01XX 110111001011
5	3 2 1 0 F E	01XX 001100100001
6	9 8 7 6 5 4	01XX 100110000111
7	F E D C B A	01XX 111111101101
8	5 4 3 2 1 0	01XX 010101000011
9	B A 9 8 7 6	01XX 101110101001

Alt.6x10x4 MODE 10 Lines Per Row CELL N

01XX 001000010000
01XX 100001110110
01XX 111011011100
01XX 010000110010
01XX 101010011000
01XX 000011111110
01XX 011001010100
01XX 110010111010
01XX 001000010000
01XX 100001110110

CELL N+1

Figure 3.21 DRCS memory coding example

Mode	B15 B12	B11 .. B8	B7 ... B4	B3 ... B0	Comments
					Two memory cells
Alternative 6x12x4	0 1 xx	D5 A3 ... A0	D4 A3 ... A0	D3 A3 ... A0	First cell defines pixels D5, D4 and D3
		D2 A3 ... A0	D1 A3 ... A0	D0 A3 ... A0	Second cell defines pixels D2, D1 and D0

LINE NO.	DISPLAY	MEMORY DATA MODE
0	5 4 3 2 1 0	01XX 010101000011
1	B A 9 8 7 6	01XX 101110101001
2	1 0 F E D C	01XX 000100001111
3	7 6 5 4 3 2	01XX 011101100101
4	D C B A 9 8	01XX 110111001011
5	3 2 1 0 F E	01XX 001100100001
6	9 8 7 6 5 4	01XX 100110000111
7	F E D C B A	01XX 111111101101
8	5 4 3 2 1 0	01XX 010101000011
9	B A 9 8 7 6	01XX 101110101001
10	1 0 F E D C	01XX 000100001111
11	7 6 5 4 3 2	01XX 011101100101

Alt.6x12x4 MODE 12 Lines Per Row CELL N

01XX 001000010000
01XX 100001110110
01XX 111011011100
01XX 010000110010
01XX 101010011000
01XX 000011111110
01XX 011001010100
01XX 110010111010
01XX 001000010000
01XX 100001110110
01XX 111011011100
01XX 010000110010

CELL N+1

Figure 3.22 DRCS memory coding example

3.7 FULL-FIELD DRCS

EUROM has a facility that allows every character position on the screen to contain a different character matrix; this is Full-Field DRCS. Its most important application is for bit-map displays. One chapter of DRCS memory can contain:

```
  51   12x10x2 or 6x10x4 characters,
 102   12x10x1 or 6x10x2 characters,
or
 204   6x10x1 or 6x5x2 characters,
```

or any combination of these. As EUROM provides two DRCS display chapter pointer registers, up to 408 different DRCS characters can be displayed. This would be insufficient to cover the whole screen.

Within EUROM, the Full-Field DRCS mode is achieved by simply enabling a counter that is loaded from the DRCS chapter pointer 'A' register (R7) and clocking at a 2 µs rate during the display time of one tv line (see Figure 3.23).

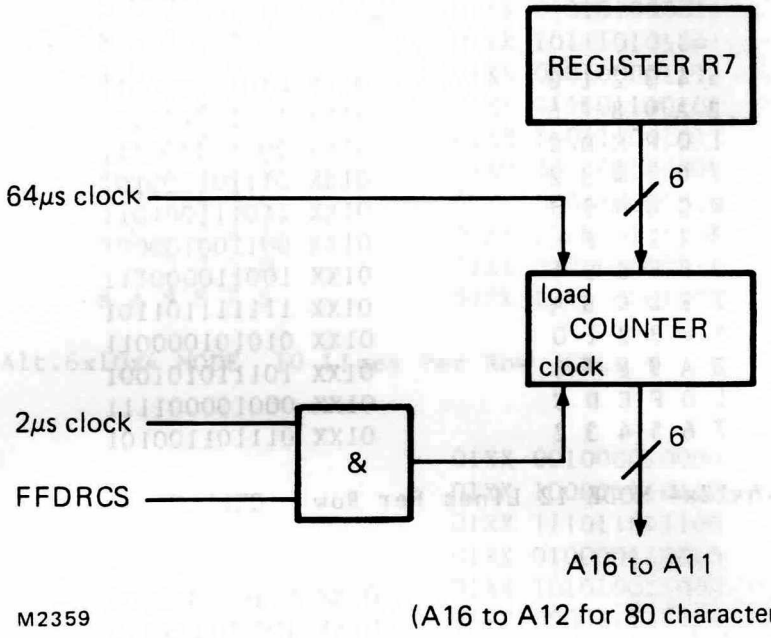


Figure 3.23 Full-Field DRCS Mechanism

A new chapter of DRCS memory is thus introduced every two columns of display. Two display columns multiplied by 25 rows gives 50 character positions, and since the most memory intensive DRCS modes provide 51 different character matrices per chapter, it can be seen that Full-Field DRCS allows the whole screen to be covered with fully random dot data using any of the available DRCS display modes.

With Full-Field DRCS enabled, all the basic display modes (i.e. Stack, Explicit Fill and 80 characters/row) may be used, and all the on-chip ROM-based characters and all the display attributes are still available. This permits displays that may be a mixture of ROM-based characters and 'bit map' display. It should be noted that when the 80 characters/row mode is used with Full-Field DRCS, the display chapter is apparently incremented every 8 character columns. (See Section 4.7, R4 Bit 4 - Full-Field DRCS).

Bit-map example

The 'alternative' 6x10x4 DRCS mode is especially suitable for Full-Field DRCS 'bit-map' applications. In this mode, each character matrix is stored as two 10-word memory cells, with the first memory cell providing dots D5, D4 and D3, and the second memory cell providing dots D2, D1, and D0.

	MSB			LSB
Alternative 6x10x4	0 1 xx	D5 A3 A2 A1 A0	D4 A3 A2 A1 A0	D3 A3 A2 A1 A0
		D2 A3 A2 A1 A0	D1 A3 A2 A1 A0	D0 A3 A2 A1 A0

(A3..A0 are Colour Map addresses)
(Colour Map A4 from R3/1)

If the character memory is organised as shown below, there can be a simple relationship between the (x,y) pixel address and the DRCS memory address:

Column	0 1	2 3	4 5	6 7	.. ..	.. ..	38 39
Row 0	0 2	0 2	0 2	.. ..	.. ..	.. ..	0 2
1	4 6	4 6	4 6	.. ..	.. ..	.. ..	4 6
2	8 10	8 10	8 10	.. ..	.. ..	.. ..	8 10
3	12 14	12 14	12 14	.. ..	.. ..	.. ..	12 14
4	16 18	16 18	16 18	.. ..	.. ..	.. ..	16 18
5	20 22	20 22	20 22	.. ..	.. ..	.. ..	20 22
.	.. ..	.. ..	.. ..	.. ..	.. ..	.. ..	.. ..
.	.. ..	.. ..	.. ..	.. ..	.. ..	.. ..	.. ..
.	.. ..	.. ..	.. ..	.. ..	.. ..	.. ..	.. ..
.	.. ..	.. ..	.. ..	.. ..	.. ..	.. ..	.. ..
22	88 90	88 90	88 90	.. ..	.. ..	.. ..	88 90
23	92 94	92 94	92 94	.. ..	.. ..	.. ..	92 94
24	96 98	96 98	96 98	.. ..	.. ..	.. ..	96 98
Chapter: offset	(R7)+0	(R7)+1	(R7)+2	.. ..	.. ..	.. ..	(R7)+19

With this memory organisation of character codes, the corresponding organisation of DRCS memory cells (10-words/cell) is as shown below:

Column	0 1	2 3	4 5	6 7	.. ..
Row 0	0 1 2 3	0 1 2 3	0 1 2 3	.. ..	
1	4 5 6 7	4 5 6 7	4 5 6 7	.. ..	
2	8 9 10 11	8 9 10 11	8 9 10 11	.. ..	
3	12 13 14 15	12 13 14 15	12 13 14 15	.. ..	
.	.. ..	.. ..	.. ..	.. ..	
.	.. ..	.. ..	.. ..	.. ..	

Considering one line of row 0, the 12 pixels that occur in two character-columns are taken in simple order from DRCS memory cells 0, 1, 2, and 3, with 3 pixels coming from each word of memory; that is, for row 0 of the above table:

Chapter_cell_number = (x DIV 3) MOD 4

The calculation process is best described using pseudocode as follows:

```
PROCEDURE bit_map (x, y,
                   (returning) Word_address,Chapter_offset,
                               Dot_withing_word );

[--    This procedure translates an x-y pixel location into a EUROM DRCS memory
 --    address when EUROM is used in its Full Field DRCS mode.  This method
 --    assumes the use of the 'Alternative' 6x10x4 DRCS display mode.
 --
 --    Given an (x,y) pixel location in the bit-map.
 --
 --
 --                  0 <= x <= 239
 --                  0 <= y <= 249
 --
 --    the procedure calculates the DRCS memory address in the form:
 --
 --                  Chapter_offset
 --                  Word_address
 --                  Dot_within_word
 --
 --]


 VAR
    x                                [   0 <=        x          <= 239    ]
    y                                [   0 <=        y          <= 249    ]
    Word_address                     [   0 <=  Word_address     <= 1022   ]
    Dot_within_word                  [   0 <=  Dot_within_word  <= 2      ]
    Chapter_offset                   [   0 <=  Chapter_offset   <= 19     ]
    Chapter_cell_num                 [   0 <=  Chapter_cell_num <= 99     ]
    Row                              [   0 <=        Row        <= 24     ]
    Line                             [   0 <=        Line       <= 9      ]

BEGIN
    Dot_within_word := x MOD 3;
    Chapter_offset  := x DIV 12;
    Row  := y DIV 10:
    Line := y MOD 10:
    Chapter_cell_num:= ( x DIV 3) MOD 4      + 4 * Row;
    Word_address    := Chapter_offset * 2048 + 2 * ( 10*Chapter_cell_num + line )
END;

[ End-of-procedure ]
```

SECTION 4

EUROM REGISTER MAP

4.1 USE OF REGISTER MAP

The EUROM Register Map is used for defining the detailed operation and data contents of the chip. To the external controlling processor, the register map appears to be an 8-bit wide RAM with six address lines and four control lines. The data lines are connected to the most significant part of the bus, i.e. to pins 11 (D8/A9) to 4 (D15/A16) and the address lines are connected to the least significant part of the bus, i.e. to pins 19 (D0/A1) to 14 (D5/A6). (D6/A7 and D7/A8 are not used for Register Map address decoding.)

The four control lines are:

(1) Register Enable ($\overline{RE}$, pin 3)

(2) Read/Write (R/$\overline{W}$, pin 39)

(3) Upper Data Strobe ($\overline{UDS}$, pin 34)

(4) Data Transfer Acknowledge (($\overline{DTACK}$), pin 36)

$\overline{LDS}$ and $\overline{AS}$ are not used for controlling the Register Map.

The detailed timings of the microprocessor read or write for EUROM are given in the published data. However, it should be noted that there is an internal synchronisation period following a microprocessor write to EUROM during which subsequent attempted writes (or reads) will be temporarily suspended (by a delayed assertion of $\overline{DTACK}$). Normal $\overline{DTACK}$ timing will be observed if 500ns is allowed from the rising edge of $\overline{UDS}$ (with Write) to the falling edge of the next $\overline{UDS}$.

4.2 REGISTER MAP CONVENTIONS

In general the conventions adopted for EUROM in naming and numbering are logical and correspond where appropriate with the European specification mentioned in Section 1.1. There are however two areas where physical logic requirements do not correspond with transmission protocols, as follows.

The first difference is in display row and column addressing. For EUROM these start with 0. EUROM's rows are numbered 0 to 24 and its columns are numbered 0 to 39 (or 79).

The second difference relates to colour map addressing. For EUROM the first colour look-up table (colour map locations 0 - 7) contains Transparent, whereas CEPT Videotex uses the second colour look-up table, i.e. EUROM's 'Intensity' bit when set is intended to refer to full intensity colours. Tables 4.1 and 4.2 show the 59 registers available and their designated functions.

Table 4.1 Register map (Registers 0 - 17)
(See also separate 'pull-out' at rear of document)

	B7	B6	B5	B4	B3	B2	B1	B0	
0	VCS/OSCO	F1/6	FS/DDA	TCS IN/OUT	10/12 LINES/ROW	PL ENABLE	ACCEPT VCS	INTER-LACE	SYNC REGISTER
1						T2	T1	T0	TEST REGISTER
2				T4	T3	T2	T1	T0	DMA "WARNING TIME" (T in uSec)
3	TV EN.	TEXT EN.	WBEN				PAL. (COL. DRCS)	FLASH INHIB.	DISPLAY COMMAND REGISTER 1
4				FULL FLD. DRCS	WHITE BUTTON	SCREEN REVEAL	M1	M0	DISPLAY COMMAND REGISTER 2 (Note 1.)
5			N5	N4	N3	N2	N1	N0	DISPLAY START (N IN LINES)
6			A16	A15	A14	A13	A12	A11	DISPLAY PAGE CHAPTER POINTER
7			A16	A15	A14	A13	A12	A11	DRCS CHAPTER 'A'
8			A16	A15	A14	A13	A12	A11	DRCS CHAPTER 'B'
9			C5	C4	C3	C2	C1	C0	CURSOR COLUMN
10	T2	T1	T0	F4	F3	F2	F1	F0	CURSOR CHARACTER CODE AND ATTRIBUTE REGISTERS
11		FLEN	C	B4	B3	B2	B1	B0	
12	U	P6	P5	P4	P3	P2	P1	P0	
13	A1	A0		D4	D3	D2	D1	D0	D-CLUT
14				A4	A3	A2	A1	A0	ADDRESS) COMPLOT
15					D3	D2	D1	D0	DATA (R))
16					D3	D2	D1	D0	DATA (G))
17					D3	D2	D1	D0	DATA (B))

Table 4.2 Register map (Registers 18 - 58)

	B7	B6	B5	B4	B3	B2	B1	B0		
18									UNUSED	
(									)	
(									)	
(									)	
32									UNUSED	
R33	DIS--ABLE RAM ACCESS	DIS--ABLE TEXT DISP.	DIS--ABLE DOUBLE HEIGHT	R4	R3	R2	R1	R0	SCAN ROW 0	)
R(	"	"	"	"	"	"	"	"		) SCROLL MAP
R(	"	"	"	"	"	"	"	"		)
R57	DRA	DTD	DDH	R4	R3	R2	R1	R0	SCAN ROW 24	)
R58	DRA	1	X	X	X	X	X	X	SCAN ROW ROW '25'	)

Note (1) M0 and M1 determine the fundamental display mode:

M1	M0	MODE
0	0	Stack Mode
0	1	This Combination is Not Allowed
1	0	Explicit Fill Mode
1	1	80 Characters/Row Mode

Note (2) R = Readable Register (R33 to R58)

Note (3) Scan Row '25' corresponds to the Lower Board Area (LBA) at the bottom of the display.

Note (4) Individual register bits may be referred to as R register bit X. i.e. White Button is R4 bit 3.

4.3 REGISTER 0 (R0) - SYNC REGISTER

This **write-only** register controls the basic timing modes of EUROM. The register contents are defined as follows:

SYNC REGISTER

B7	B6	B5	B4	B3	B2	B1	B0
VCS/ $\overline{\text{OSCO}}$	F1/ $\overline{6}$	FS/ $\overline{\text{DDA}}$	TCS IN/ $\overline{\text{OUT}}$	$\overline{10}$/12 LINES/ ROW	$\overline{\text{PL}}$ ENABLE	ACCEPT VCS	INTER-LACE

R0 BIT 0 - INTERLACE

This bit controls the type of $\overline{\text{TCS}}$ (Text Composite Sync, pin 32) output waveform, selecting interlace or non-interlace, when EUROM is acting as sync master (see R0 Bit 4).

If R0 bit 0 = 0, $\overline{\text{TCS}}$ is non-interlaced - even fields of 312 lines are generated repetitively.

If R0 bit 0 = 1, $\overline{\text{TCS}}$ corresponds to the normal 625 line/50Hz interlaced broadcast standard.

When EUROM is not acting as sync master, then R0 bit 0 is a "dont care".

R0 BIT 1 - ACCEPT VCS

This bit controls the signal source for internal field sync integrator.

If R0 bit 1 = 0, pin 32 ($\overline{\text{TCS}}$) is used.

If R0 bit 1 = 1, pin 31 (VCS) is used.

R0 BIT 2 - $\overline{\text{PL ENABLE}}$

This bit controls the PL (Phase Lock) part of the sandcastle waveform (SAND, pin 28).

If R0 bit 2 = 0, PL is enabled and when a SAA5230 device is included in the system, enables the SAA5230 phase-locked loop to lock (in a horizontal sync sense) onto an incoming source of composite video (e.g. broadcast or VLP).

If R0 bit 2 = 1, the PL part of the sandcastle waveform is disabled and SAA 5230 will free run its 6MHz output clock (see Figure 4.1).

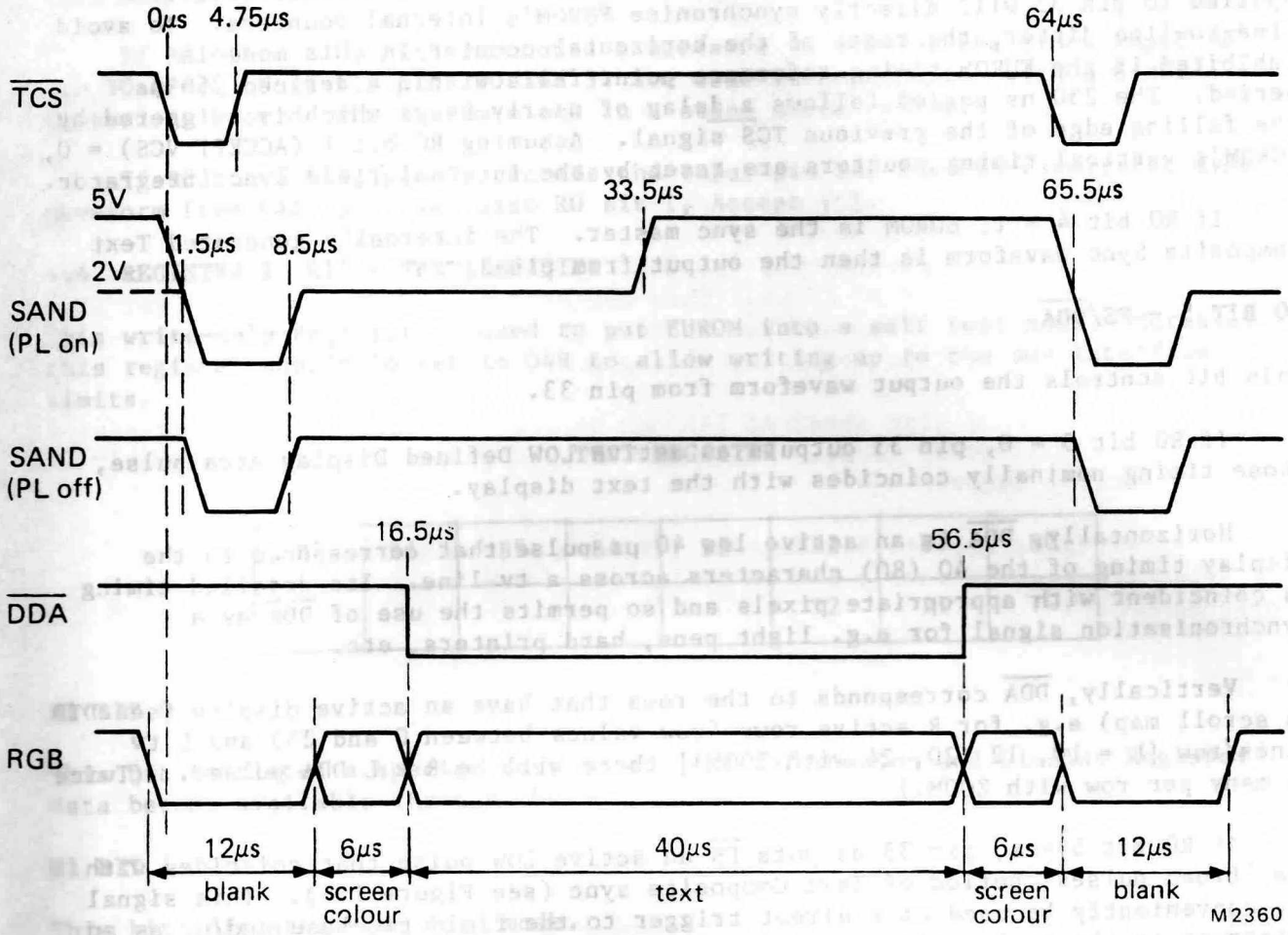


Figure 4.1 Line timing waveforms

R0 BIT 3 - $\overline{10}$/12 LINES/ROW

This bit controls the number of scan lines used for each row of text display.

If R0 bit 3 = 0, 10 scan lines per row are used.

If R0 bit 3 = 1, 12 scan lines per row are used.

Note: When 12 lines per row are used it is not possible to display 25 rows of text in one field of the 625 line/50Hz scanning system. The exact number of rows displayable depends on the retrace time and degree of overscan of the monitor used.

R0 BIT 4 - TCS IN/OUT

This bit controls whether EUROM is the sync master or sync slave in a system.

If R0 bit 4 = 0, EUROM becomes a sync slave - an external $\overline{TCS}$ waveform applied to pin 32 will directly synchronise EUROM's internal counters. To avoid line-to-line jitter, the reset of the horizontal counter in this mode is inhibited if the EUROM timing reference point falls within a defined 250 ns period. The 250 ns period follows a delay of nearly 64 µs which is triggered by the falling edge of the previous $\overline{TCS}$ signal. Assuming R0 bit 1 (ACCEPT VCS) = 0, EUROM's vertical timing counters are reset by the internal field sync integrator.

If R0 bit 4 = 1, EUROM is the sync master. The internally generated Text Composite Sync waveform is then the output from pin 32.

R0 BIT 5 - FS/$\overline{DDA}$

This bit controls the output waveform from pin 33.

If R0 bit 5 = 0, pin 33 outputs an active LOW Defined Display Area pulse, whose timing nominally coincides with the text display.

Horizontally, $\overline{DDA}$ is an active low 40 µs pulse that corresponds to the display timing of the 40 (80) characters across a tv line. Its detailed timing is coincident with appropriate pixels and so permits the use of $\overline{DDA}$ as a synchronisation signal for e.g. light pens, hard printers, etc.

Vertically, $\overline{DDA}$ corresponds to the rows that have an active display (see DTD in scroll map) e.g. for R active rows (row values between 0 and 25) and L tv lines/row [L = 10, 12 (20, 24 with ZOOM)] there will be R x L DDA pulses. (Twice as many per row with ZOOM.)

If R0 bit 5 = 1, pin 33 outputs $\overline{FS}$ an active LOW pulse that coincides with the 'broad pulses' period of Text Composite sync (see Figure 1.6). This signal may conveniently be used as a direct trigger to the field timebase and/or as an interrupt to the system microprocessor to initiate scrolling, colour map redefinition or other routines that are best performed during the non-display time.

R0 BIT 6 - F1/$\overline{6}$

This bit controls the output clock from pin 29.

If R0 bit 6 = 0, pin 29 outputs a 6 MHz squarewave which has a defined relationship with EUROM's bus cycle timing. This permits interleaving of EUROM and microprocessor memory cycles.

If R0 bit 6 = 1, pin 29 outputs a 1 MHz squarewave suitable for driving peripheral devices, e.g. SAA5070.

Note: The 500 ns HIGH or LOW period of the 1 MHz square-wave may be momentarily reduced to a nominal 83 ns in slave sync mode at the moment of resynchronisation.

R0 BIT 7 - VCS/OSC0

This bit controls whether pin 31 is an input or an output. It is reset to 0 following power up.

If R0 bit 7 = 0, pin 31 outputs a buffered version of the clock input on pin 30 (F6). Normally pins 30 and 31 are used in conjunction with a 6 MHz crystal to provide the system clock in a stand alone system.

If R0 bit 7 = 1, pin 31 becomes the input pin for the VCS (separated sync) waveform from SAA5230. See also R0 bit 1, Accept VCS.

4.4 REGISTER 1 (R1) - TEST REGISTER

This **write-only** register is used to put EUROM into a self test mode. Normally this register should be set to 04H to allow writing up to the bus interface limits.

TEST REGISTER

B7	B6	B5	B4	B3	B2	B1	B0
					T2	T1	T0

R1 BIT 0

This bit initiates a special test mode. ROM information and Display Register data become available through the bus.

R1 BIT 1

This bit initiates scan test mode.

R1 BIT 2

This bit controls writing into the Colour Map. With R1/2 = 0, data transfer can only occur during display blanking. When this bit is set, writing into the Colour Map can occur at any rate up to the limit determined by the bus interface requirements.

R1 BIT 3 to R1 BIT 7

These bits are unused.

4.5 REGISTER 2 (R2) - DMA WARNING TIME REGISTER

DMA WARNING TIME REGISTER (T in µs)

B7	B6	B5	B4	B3	B2	B1	B0
			T4	T3	T2	T1	T0

EUROM fetches two main types of data from external memory - Row Buffer data (character, attribute and screen colour information) and DRCS dot data. Row Buffer data is fetched during the 24 µs periods roughly corresponding to line flyback time (one or two periods per row depending on mode) and DRCS dot data (if any) is fetched during the 40 µs periods corresponding to the active display (Defined Display Area - 'DDA') time.

Because EUROM is a real-time display device, it must have direct access to the memory (DMA) with priority over the controlling microprocessor or other peripheral devices. To this end EUROM issues a Bus Request ($\overline{BR}$, pin 37) signal for the duration of the memory access plus an additional pre-settable advance warningtime to allow the microprocessor to release possession of the bus.

R2 is a **write-only** register used for controlling the duration of the advance warning time up to a maximum of 23 µs. This register is not automatically set to a default value following power up, but instead EUROM memory accesses are inhibited until R2 is written to by the microprocessor.

The advance warning time obtained (in µs) is nominally equal to the register contents up to a maximum of 23 µs. Register contents larger than 23 will result in negative warning times.

The actual times between Bus Request ($\overline{BR}$, pin 37) being asserted and the memory interface signals coming out of a high impedance state are summarised in Table 4.3.

Table 4.3 DMA warning times

Value loaded into Warning Time Register	Time obtained Row Buffer data	Time obtained DRCS data
0	0.5 µs + x	0 µs + x
1	1.5 µs + x	1 µs + x
:	:	:
:	:	:
22	22.5 µs + x	22 µs + x
23	23 µs + x	22.5 µs + x

x is approximately 100 ns

The Data, Address, $\overline{UDS}$, $\overline{LDS}$, R/$\overline{W}$ and $\overline{AS}$ pins return to a high impedance state after memory accessing is complete. $\overline{BR}$ is negated approximately 150 ns later for a page memory reference or approximately 650 ns later for a DRCS memory reference.

4.6 REGISTER 3 (R3) - DISPLAY COMMAND REGISTER 1

This **write-only** register is used to control various aspects of the display.

DISPLAY COMMAND REGISTER 1

B7	B6	B5	B4	B3	B2	B1	B0
TV EN.	TEXT EN.	WBEN				PAL. (COL. DRCS)	FLASH INHIB.

Table 4.4 Interpretation of display mode control bits TVEN, TEXTEN, WBEN

TV EN.	TEXT EN.	WBEN	
0	0	0	"Black Screen" function. Causes the contents of colour map location 0 to be output on the RGB pins and disables all EUROM bus activity (including Bus Requests).
0	0	1	Output Disable (OD) function. Causes the RGB and VDS pins to go high impedance. Bus activity disabled.
0	1	0	Normal Text mode with Window/Box attribute disabled.
0	1	1	Normal Text mode with Window/Box attribute enabled. Where the Window/Box attibute is active, the Screen Colour will be set to transparent.
1	0	0	TV mode. The contents of colour map location 0 is output on the RGB pins and VDS is set to 0 to enable the display of the TV video. EUROM bus activity is disabled (including Bus Request).
1	0	1	TV mode with boxes. Text will only be displayed (in boxes) where the Window/Box attribute is active. The remainder of the display will be the TV video.
1	1	0	Superimpose mode. Text will be superimposed on top of the TV video. The Window/Box attribute is disabled.
1	1	1	Superimpose mode with boxes. Text will be superimposed on top of the TV video. Boxes will be displayed where the Window/Box attribute is active.

R3 BIT 0 - FLASH INHIBIT

This bit is used to inhibit the effects of any flash attribute. Typically it would be used when taking a hard copy of the screen image.

If R3 bit 0 = 0, flashing is enabled.

If R3 bit 0 = 1, flashing is inhibited.

R3 BIT 1 - PALETTE (COLOUR DRCS)

This bit is used to provide the most significant address bit to the on-chip colour map when working with 16 colour (4 bit/dot) DRCSs.

If R3 bit 1 = 0, 16 colour DRCSs use locations 0 - 15 of the colour map and so may include 'transparent'.

If R3 bit 1 = 1, 16 colour DRCSs use locations 16 - 31 of the colour map.

R3 BIT 2 TO R3 BIT 4

These bits are unused.

R3 BIT 5 - WINDOW OR BOX ENABLE

This bit can be used to enable/disable the 'Window' attribute in text mode or the 'Box' attribute in tv or superimpose modes. In conjunction with R3 bit 6 (Text Enable) and R3 bit 7 (TV Enable) it may also be used for controlling bus activity and the state of the RGB and $\overline{\text{VDS}}$ outputs. See Screen Attributes functional description in Table 4.4.

If R3 bit 5 = 0, the Window or Box attribute is disabled.

If R3 bit 5 = 1, the Window or Box attribute is enabled.

R3 BIT 6 - TEXT ENABLE

This bit controls the display of text on the screen. (Also see Sections 1.8 and 1.11).

If R3 bit 6 = 0, Text Display is disabled (If WBEN = 0. Otherwise text in boxes is displayed).

If R3 bit 6 = 1, Text Display is enabled.

R3 BIT 7 - TV ENABLE

This bit is used to control the display of TV (or 'other') video on the screen. (Also see Sections 1.8 and 1.11).

If R3 bit 7 = 0, TV display is disabled

If R3 bit 7 = 1, TV display is enabled.

4.7 REGISTER 4 (R4) - DISPLAY COMMAND REGISTER 2

This **write-only** register controls various aspects of the display as follows.

DISPLAY COMMAND REGISTER 2

B7	B6	B5	B4	B3	B2	B1	B0
			FULL FLD. DRCS	WHITE BUTTON	FULL SCREEN REVEAL	M1	M0

R4 BIT 0 - DISPLAY MODE BIT 0, M0

This bit controls the number of columns to be used for the display. (See notes to Table 4.2.)

If R4 bit 0 = 0, 40 columns per row are used.

If R4 bit 0 = 1, 80 columns per row are used.

R4 BIT 1 - DISPLAY MODE BIT 1, M1

This bit controls the interpretation of page memory data when Row Buffer filling. (See notes to Table 4.2.)

If R4 bit 1 = 0, page data is interpreted as being Stack coded. 80 bytes (40 words) of page data are fetched from external RAM per display row during one 24 µs memory access period.

If R4 bit 1 = 1, page data is interpreted as being 'explicit' - 160 bytes (80 words) of page data are fetched from external RAM per display row during two 24 µs memory access periods, the page data being transferred directly to the Row Buffer without modification.

The combination RM0 = 1 and RM1 = 0 is not allowed.

R4 BIT 2 - FULL SCREEN REVEAL

This bit controls the display of concealed text.

If R4 bit 2 = 0, the conceal attribute is enabled, i.e. concealed text is not legible.

If R4 bit 2 = 1, the conceal attribute is disabled.

See also Section 1.8 (Row Reveal).

R4 BIT 3 - WHITE BUTTON

This bit sets screen colour to Black and all attributes to their default values as follows.

White Foreground

Transparent Background

Not Flashing

Normal Size

Not Lined

Not Inverted

Not Concealed

No Window/Box

Character table and lock bit (Stack code 1,1,0,L,T2,T1,T0) are unaffected. White Button is active in Stack, Explicit Fill and 80 characters/row modes.

If R4 bit 3 = 0, attributes are enabled as selected.

If R4 bit 3 = 1, attributes are set (internally) to their default values.

R4 BIT 4 - FULL FIELD DRCS

This bit controls the incrementing of the DRCS 'A' chapter pointer (R7). The contents of R7 are transferred to a 6-bit counter which is incremented after every 2 character columns during active display time. (In the 80 characters/row mode the LSB of this counter is not brought out as an external address, so here a 5-bit chapter pointer increments after every 8 character columns.)

If R4 bit 4 = 0, chapter pointer incrementing is inhibited.

If R4 bit 4 = 1, chapter pointer incrementing is enabled.

See also Section 3.4 (DRCS Addressing (40 Characters/Row)).

4.8 REGISTER 5 (R5) - DISPLAY START REGISTER

This **write-only** register controls the vertical start position of text on screen. Typical uses are to centre the display or achieve soft vertical scroll. (See the field timing waveforms in Section 5.4).

DISPLAY START REGISTER (N in lines)

B7	B6	B5	B4	B3	B2	B1	B0
		N5	N4	N3	N2	N1	N0

When operating in a 10 lines/row display mode, the first line of the first row of text display will be on line number (N + 15) where N represents the contents of R5. If 25 rows of text are used, vertical display centering will be achieved with N = 27.

When operating in the 12 lines/row mode the first display line will be (N + 17). Here if 21 rows of text are used, vertical display centering will be achieved with N = 24.

It should be noted that if extreme values of N are used in R5 in conjunction with zoom or the maximum number of rows/page (see R33 - R58, Scroll Map) it is possible to move the text off the screen into the Field Blanking region.

The minimum value of N permitted is decimal 3 (scroll map availability).

4.9 REGISTER 6 (R6) DISPLAY PAGE CHAPTER POINTER

The external memory directly addressable by EUROM can be considered as consisting of 64 chapters of 1024 16-bit words. R6 is a **write-only** register that defines the starting position in memory to be used for Page Data.

DISPLAY PAGE CHAPTER REGISTER

B7	B6	B5	B4	B3	B2	B1	B0
		A16	A15	A14	A13	A12	A11

In Stack mode 1024 16-bit words are needed per page so R6 can be used to address up to 64 different pages.

In Explicit Fill or 80 characters/row modes 2048 16-bit words are needed per page. Here R6 bit 0 becomes a "don't care" bit and up to 32 different pages are addressable.

4.10 REGISTER 7 (R7) - DRCS CHAPTER 'A' POINTER

This **write-only** register defines the starting position in memory to be used for DRCS data. When R4 bit 4 = 1 (Full Field DRCS) this pointer increments in sympathy with the horizontal scan, enabling Full Field DRCS mode (see Section 3.7).

DRCS CHAPTER 'A' REGISTER

B7	B6	B5	B4	B3	B2	B1	B0
		A16	A15	A14	A13	A12	A11

In 40 characters/row mode, 1024 16-bit words are needed per Chapter of DRCS data, so R7 can be used to address up to 64 different DRCS sets.

In 80 characters/row mode, 2048 16-bit words (or 2K8 if 1 bit/dot only is used) are needed per chapter of DRCS data, so here R7 bit 0 becomes a "don't care" bit and up to 32 different DRCS sets are addressable.

4.11 REGISTER 8 (R8) - DRCS CHAPTER 'B' POINTER

This **write-only** register defines the starting position in memory for DRCS data. This pointer is not affected in Full Field DRCS mode and is not available in 80 characters/row mode. R8 can be used to address up to 64 different DRCSs.

DRCS CHAPTER 'B' POINTER

B7	B6	B5	B4	B3	B2	B1	B0
		A16	A15	A14	A13	A12	A11

4.12 REGISTER 9 (R9) - CURSOR COLUMN

This **write-only** register defines the character column to be used for cursor display (Stack mode only). The cursor row is defined by the screen colour bytes in external RAM. (See also Sections 1.8 and 2.2).

The cursor is only active, in Stack Mode, between columns 0 and 39. Any value outside this range will disable the cursor display and this feature can be used, in software, for this purpose.

CURSOR COLUMN

B7	B6	B5	B4	B3	B2	B1	B0
		C5	C4	C3	C2	C1	C0

4.13 REGISTER 10 (R10) - CURSOR CHARACTER TABLE AND FOREGROUND COLOUR

This **write-only** register determines the character table and foreground colour to be used for cursor display in Stack Mode. The contents of this register or even the presence of a cursor do not affect the display of the rest of the cursor row.

CURSOR CHARACTER TABLE AND FOREGROUND COLOUR REGISTER

B7	B6	B5	B4	B3	B2	B1	B0
T2	T1	T0	F4	F3	F2	F1	F0

4.14 REGISTER 11 (R11) - CURSOR FLASH ENABLE, CONCEAL AND BACKGROUND COLOUR

This **write-only** register determines the Flash, Conceal and Background Colour attributes to be used for cursor display in Stack Mode. The contents of this register or even the presence of a cursor do not affect the display of the rest of the cursor row.

CURSOR FLASH ENABLE, CONCEAL AND BACKGROUND COLOUR REGISTER

B7	B6	B5	B4	B3	B2	B1	B0
	FLEN	C	B4	B3	B2	B1	B0

R11 BIT 0 TO R11 BIT 4 - CURSOR BACKGROUND COLOUR

These bits determine the Background colour to be used for cursor display in Stack Mode.

R11 BIT 5 - CURSOR CONCEAL

This bit determines the Conceal attribute to be used for cursor display in Stack Mode.

If R11 bit 5 = 0, the cursor will display normally even if the page memory contents are such that the Conceal attribute is set at or pertaining to the cursor position.

If R11 bit 5 = 1, the cursor is concealed and normally only the cursor Background colour will be visible at the cursor position. In conjunction with suitable software this bit may be used to effect a cursor flash. (See also Sections 1.8 and 2.11).

R11 BIT 6 - CURSOR FLASH ENABLE

This bit enables the Flash attributes defined in page memory that are set at or pertaining to the cursor position in Stack Mode.

If R11 bit 6 = 0, the page memory Flash attributes for the cursor position are disabled - the cursor does not flash.

If R11 bit 6 = 1, the page memory Flash attributes for the cursor position are used for cursor display.

4.15 REGISTER 12 (R12) - CURSOR LINING AND CHARACTER CODE

This **write-only** register determines the Lining attribute and character code to be used for cursor display. The contents of this register or even the presence of a cursor do not affect the display of the rest of the cursor row.

CURSOR LINING AND CHARACTER CODE REGISTER

B7	B6	B5	B4	B3	B2	B1	B0
L	P6	P5	P4	P3	P2	P1	P0

R12 bit 0 to R12 bit 6 determine the character code to be used for cursor display.

R12 bit 7 determines the Lining attribute to be used for cursor display.

4.16 REGISTER 13 (R13) - DRCS-CLUT

This **write-only** register is used to define the contents of the DCLUT. There are no restrictions on when it may be written to by the microprocessor apart from the usual limitations of the bus interface (timings, $\overline{DTACK}$ etc.).

DRCS-CLUT REGISTER

B7	B6	B5	B4	B3	B2	B1	B0
A1	A0		D4	D3	D2	D1	D0

R13 bit 0 to R13 bit 4 defines the data contents of the addressed DCLUT location.

R13 bit 6 and R13 bit 7 determine which of the four DCLUT locations are addressed.

4.17 REGISTER 14 (R14) - COLOUR MAP ADDRESS

This **write-only** register is used to define which of the 32 Colour Map addresses will be written to with the data held in registers R15 to R17.

COLOUR MAP ADDRESS REGISTER

B7	B6	B5	B4	B3	B2	B1	B0
			A4	A3	A2	A1	A0

The action of writing into R14 initiates data transfer into the Colour Map. Normally, registers R15 to R17 will be written to by the controlling microprocessor before R14 is defined, although this is not mandatory.

The intended Colour Map RAM is time-shared between the display and the microprocessor, and so, to avoid corruption of the display, it may be desirable to transfer data to the Colour Map only during non-display time, that is during the 24 µs period corresponding to the line flyback.

This may be achieved with EUROM by setting R1/2 = 0, whereupon internal data transfers from registers R14 to R17 are delayed until line flyback time. With R1/2 = 0, a non-synchronised microprocessor can therefore write to the Colour Map at a maximum rate that is marginally slower than line frequency.

Normally, however, R1/2 will be set (see Register 1 details in Section 4.4) to allow writing into the Colour Map at the maximum rate allowable by the bus interface requirements.

4.18 REGISTER 15 (R15) - COLOUR MAP DATA (RED)

This **write-only** register contains the Red data that will be transferred to the Colour Map when R14 is written to.

COLOUR MAP DATA (RED) REGISTER

B7	B6	B5	B4	B3	B2	B1	B0
				D3	D2	D1	D0

4.19 REGISTER 16 (R16) - COLOUR MAP DATA (GREEN)

This **write-only** register contains the Green data that will be transferred to the Colour Map when R14 is written to.

COLOUR MAP DATA (GREEN) REGISTER

B7	B6	B5	B4	B3	B2	B1	B0
				D3	D2	D1	D0

4.20 REGISTER 17 (R17) - COLOUR MAP DATA (BLUE)

This **write-only** register contains the Blue data that will be transferred to the Colour Map when R14 is written to.

COLOUR MAP DATA (BLUE) REGISTER

B7	B6	B5	B4	B3	B2	B1	B0
				D3	D2	D1	D0

4.21 REGISTERS R18 TO R32 - UNUSED.

4.22 REGISTERS 33 TO 58 (R33 TO R58) - SCROLL MAP

The Scroll Map is an on-chip **Read/Write** RAM that is used primarily to control the row order in which page data is scanned. During the active part of a field, the Scroll Map is accessed in simple order and the contents found are used as pointers to page memory. In addition the Scroll Map contains two bits (per row) that may be used to control the quantity and disposition of rows displayed on the screen as well as one bit (per row) that may be used to disable the Double Height attribute when, for example, the subsequent row is (part of) a protected area.

SCROLL MAP REGISTERS

Reg.	B7	B6	B5	B4	B3	B2	B1	B0	
33	DIS--ABLE RAM ACCESS	DIS--ABLE TEXT DISP.	DIS--ABLE DOUBLE HEIGHT	R4	R3	R2	R1	R0	SCAN ROW 0
	"	"	"	"	"	"	"	"	
	"	"	"	"	"	"	"	"	
57	DRA	DTD	DDH	R4	R3	R2	R1	R0	SCAN ROW 24
58	DRA	1	X	X	X	X	X	X	SCAN ROW ROW '25'

The Scroll Map may be freely accessed by the microprocessor during the period following the Field Sync ($\overline{FS}$) pulse - see Field Timing Waveforms (Fig. 1.5). In addition the Scroll Map is available to the microprocessor from the final line of display in a field (i.e. the final $\overline{DDA}$ pulse) up to $\overline{FS}$. It should be noted that this latter period may not exist with certain combinations of ZOOM, DTD (see below) and 12 lines per row.

R33 - R57 BITS 0 - 4 - ROW DATA

Bits 0 to 4 of registers R33 to R57 hold the page memory row data for the scan rows 0 to 24. There is no restriction on the row data held in the Scroll Map except that contents with value outside the range 0 to 24 will cause an address wraparound within the display chapter.

R33 - R57 BIT 5 - DISABLE DOUBLE HEIGHT (DDH)

This bit, when set, may be used to disable the effects of any Double Height attribute that is set in the row in question (effective in Stack mode only). Any Double Height (Size) characters with their origin in the current row will be displayed normal height (Double Width). Double Height characters with their origin in the previous row are unaffected. Typically this bit would be used when the following row is protected or is part of a scrolling area, or when the current row is the final row of a field.

R33 - R58 BIT 6 - DISABLE TEXT DISPLAY (DTD)

This bit controls the selection of rows on which display text is enabled. Any number of text rows between 0 and 25 may be displayed but they must be in a single, contiguous block. The visual effect of non-contiguous rows can be achieved by suitable use of Transparent Background with spaces. The DTD bit also affects the display of screen colours as follows. If text is displayed normally on row n but is disabled for row n + 1 then R (33 + n) bit 6 = 0 and R (33 + n + 1) bit 6 = 1. Under these conditions the internal Scroll Map address counter is set immediately to point to R58 (corresponding to the Lower Border Area), and the effect on the screen (assuming R58 bit 7 = 0) is that the Screen Colour corresponding to the Lower Border Area is seen from row (n + 1) to the bottom of the screen inclusively. Other Scroll Map contents from R (33 + n + 1) to R57 are then ignored. See Tables 4.5 and 4.6. **R58 bit 6 must be permanently set HIGH.**

R33 - R58 BIT 7 - DISABLE RAM ACCESS (DRA)

This bit is used to control, on a row basis, whether EUROM is enabled to access external RAM for fetching Character, Attribute, Screen Colour or DRCS data. Typically it would be used in conjunction with DTD (Disable Text Display) to disable any RAM accesses on rows where no text will be displayed, thus increasing the proportion of time available for the controlling processor to have possession of the bus. On rows where text display is enabled but RAM access is inhibited, the (previous) contents of the row buffer are not overwritten with new data and thus may be (re-)used for display. It should be borne in mind that EUROM has two row buffers on chip which are used alternately for successive rows of display. See Sections 1.2 and 1.3 and Tables 4.5 and 4.6. DRCS cannot be displayed without access to external RAM.

Table 4.5 Examples of scroll map contents

DISPLAY ROW	DRA	DTD	DDH	DRA	DTD	DDH	DRA	DTD	DDH	DRA	DTD	DDH	DRA	DTD	DDH
			(1)												
0	0	0	A	0	0	A	1	1	X	1	1	X	1	1	X
1	0	0	A	0	0	A	1	1	X	1	1	X	1	1	X
2	0	0	A	0	0	A	0	0	A	1	1	X	1	1	X
.															
.															
.															
15	0	0	A	0	0	A	0	0	A	1	1	X	1	1	X
						(2)			(2)			(2)			(2)
16	0	0	A	0	0	1	0	0	1	0	0	1	1	0	1
17	0	0	A	X	1	X	X	1	X	X	1	X	X	1	X
18	0	0	A	X	X	X	X	X	X	X	X	X	X	X	X
.															
.															
.															
23	0	0	A	X	X	X	X	X	X	X	X	X	X	X	X
			(2)												
24	0	0	1	X	X	X	X	X	X	X	X	X	X	X	X
(3)															
LBA	0	1	-	0	1	-	0	1	-	0	1	-	1	1	-
			(4)			(5)			(6)			(7)			(8)

DRA - Disable RAM Access, DTD - Disable Text Display, DDH - Disable Double Height

Notes:

(1) 'Active'. May be set/cleared at will to suit e.g. Protected Area.

(2) Normally Double Height is disabled in last row of text display, so DDH should be set at this position.

(3) UBA = Upper Border Area, i.e. above Row 0.
LBA = Lower Border Area, i.e. below Row 24.

(4) Normal 25 rows of display.

(5) Text in row 0 to 16 only.

(6) Text in row 2 to 16 only.

(7) Single row (16) of text with RAM access enabled.

(8) Single text row with 'no' RAM access i.e. RAM only accessed following Field Sync for Upper Border Area Screen Colour. (See Field Timing Waveforms in Section 5.4.)

Table 4.6 Display corresponding to Table 4.5

	(3)	(4)	(5)	(6)	(7)
UBA	'SC -1'	'SC -1'			
0	SC 0 (1)	SC 0	"SC -1"		
1	SC 1	SC 1		"SC -1"	"SC -1"
2	SC 2	SC 2	SC 2		
. . .	. .	. .	. .		
15	SC 15	SC 15	SC 15		
16	SC 16	SC 16	SC 16	SC 16	"SC -1"
17	SC 17				
18	SC 18				
. .	. .	"SC 25" (2)	"SC 25" (2)	"SC 25" (2)	"SC -1"
23	SC 23				
24	SC 24				
LBA	"SC 25"				

Notes:

(1) SC = Screen Colour

(2) Screen Colour 16 (SC 16) would be seen here if RAM Access were disabled for the lower Border Area (i.e. if R58 bit 7 = 1).

(3) Normal 25 rows display.

(4) Text in rows 0 to 16 only.

(5) Text in rows 2 to 16 only.

(6) Single row with RAM Access Enabled.

(7) Single text row with 'no' RAM Access i.e. RAM only accessed following field sync. for Upper Border Area Screen Colour. (See Sections 1.7 and 5.2).

SECTION 5

EUROM APPLICATION DETAILS

5.1 PINNING

EUROM operates from a 5 V supply, the common earth pin being VSS and the +5 V pin VDD. The pinning details are shown in Table 5.1.

Table 5.1 EUROM pinning details

Pin No.	Name	I/O	Comments
1	VSS1	-	0 V
2	$\overline{\text{BUFEN}}$	I	Buffer Enable. 8 bit "link-through" buffer for 8 bit microprocessor control.
3	$\overline{\text{RE}}$	I	"Register Enable" Enables A1 - A6 & $\overline{\text{UDS}}$ as inputs, D8 - D15 as I/O's.
19 - 4	D0 - D15 A1 - A16	I/O	Multiplexed address and data bus. 500 ns cycle time. These pins also function as a "link-through" 8 bit buffer.
20	VSS2	-	0 V
21	REF	I	Analogue Reference Input
24, 23, 22	RGB	O	Analogue Outputs. Signals are "gamma-corrected".
25	$\overline{\text{VDS}}$	O	Video Data Switch. Dot, Screen (Row), Box and Window Video Data Switching Output for use when video signal from tv, laser disk player etc, or alpha-photographic "layer" is present. This signal will interface directly to a standard colour decoder chip. Low for tv display, high for text.
26	$\overline{\text{OD}}$	I	Output Disable. RGB and $\overline{\text{VDS}}$ go high impedance. Useable at dot rate.
27	CLKO	O	12 MHz clock output for hard copy dot synchronisation. (Referenced to output dots.)
28	SAND	O	Sandcastle feedback to SAA5230 or other circuit when display must be locked to input video such as laser disk player. The PL part of sandcastle can be disabled so as to free-run the SAA5230 phase-locked loop.

Table 5.1 EUROM pinning details (continued)

Pin No.	Name	I/O	Comments
29	F1/6	O	1 or 6 MHz output clock for driving e.g. SAA5070 (LUCY). The 6 MHz timing signal has a specified timing relationship with $\overline{UDS}$, simplifying e.g. EUROM/microprocesor memory interleave applications.
30	F6	I	6 MHz clock input from e.g. SAA5230. Internal a.c. coupling.
31	VCS/OSCO	I/O	Video Composite Sync input from e.g. SAA5230 is used as a vertical display timing phase reference when locking to input video such as laser disk player. Otherwise stand alone fixed frequency oscillator output used in conjunction with F6 input.
32	$\overline{TCS}$	I/O	Text Composite Sync input or output depending on master or slave status.
33	$\overline{FS/DDA}$	O	Field sync pulse or "Defined Display Area" flag. (Referenced to output dots.)
34	$\overline{UDS}$	I/O	Upper Data Strobe
35	$\overline{LDS}$	O	Lower Data Strobe
36	$\overline{DTACK}$	O	Data Transfer Acknowledge (Open drain)
37	$\overline{BR}$	O	Bus Request. Output to microprocessor (Open drain).
38	$\overline{AS}$	O	Address Strobe. Output to external address latches.
39	R/$\overline{W}$ ($\overline{S}$/R)	I/O	Read/Write. This pin also serves as a "send/receive" for the "link-through" buffer.
40	VDD	-	5 V supply

5.2 MICROPROCESSOR AND RAM INTERFACING

Three types of data transfer take place at the bus interface:

(1) EUROM fetches data from the external display memory.

(2) The microprocessor reads from the Scroll Map registers, or writes to all of EUROM's internal register map.

(3) The microprocessor accesses the external display memory independently.

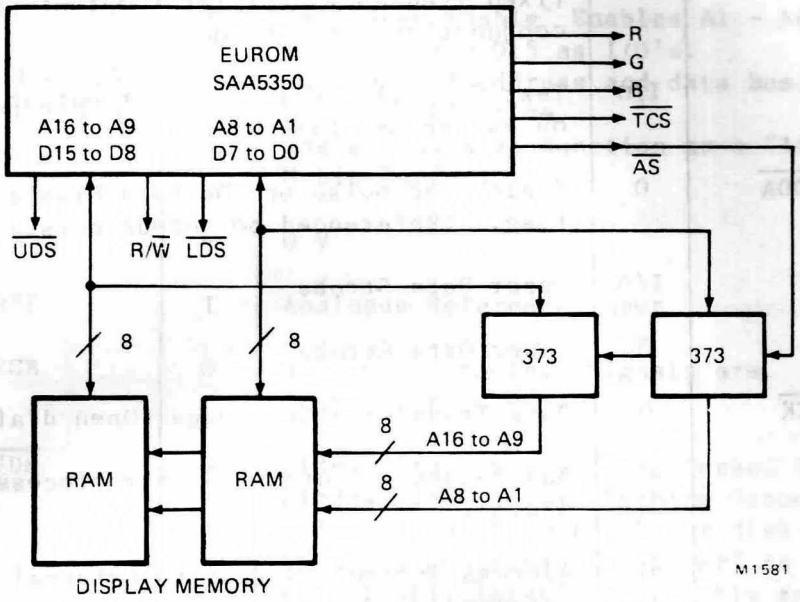
EUROM access to display memory

EUROM accesses the external display memory via a 16-bit multiplexed address and data bus with a 500 ns cycle time. Figure 5.1 shows a rudimentary RAM interface circuit and bus timing diagram.

When EUROM accesses the display memory, its Address Strobe signal $\overline{AS}$ flags the bus cycle and writes the address into the '373' latches. The display RAMs are enabled by Upper Data Strobe ($\overline{UDS}$) and Lower Data Strobe ($\overline{LDS}$) respectively.

Note that $\overline{UDS}$ and $\overline{LDS}$ are always asserted together to fetch a 16-bit word. Thus EUROM never fetches a single byte from memory.

The Read/Write control signal, R/$\overline{W}$, is included in the cycle for completeness although EUROM only 'reads' the display memory.



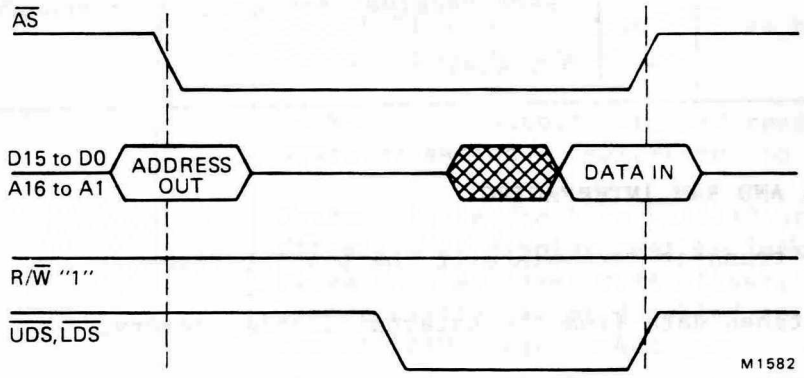


Figure 5.1 EUROM access to display memory

Although the EUROM data bus is 16 bits wide, the data fetched is often considered to exist in terms of bytes. Even-numbered bytes are on the left or upper (most significant) part of the bus, as shown in Figure 5.2. The word addresses are numerically the same as the upper byte they contain. There are no odd-numbered word addresses.

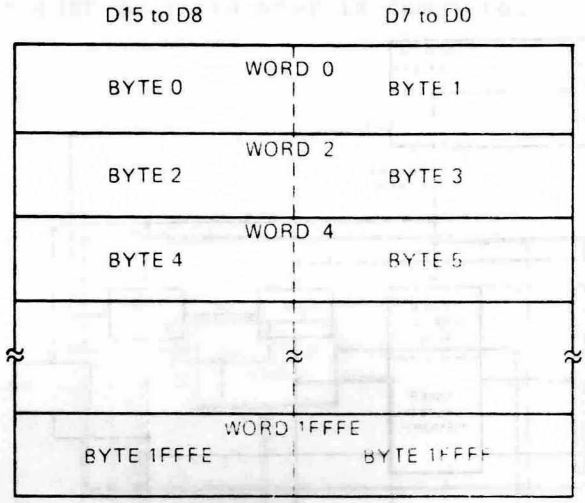


Figure 5.2 EUROM's display memory word/byte organisation

Display memory transfer timing (Bus Request)

Because EUROM is a real-time display device, it must have direct access to the display memory with priority over the controlling microprocessir or other peripheral devices. To achieve this, EUROM issues a Bus Request $\overline{BR}$ signal (not shown on Figure 5.1) for the duration of the memory access plus a programmable advance warning time to allow the microprocessor to complete its current bus cycle.

In systems where the microprocessor's bus and EUROM's bus form a 'connected' system, as in Figures 5.3, BR may be used to suspend all microprocessor activity so that EUROM acts as a dedicated DMA controller. The warning time between the assertion of BR and the beginning of EUROM's bus activity is programmable from 0 to 23 µs.

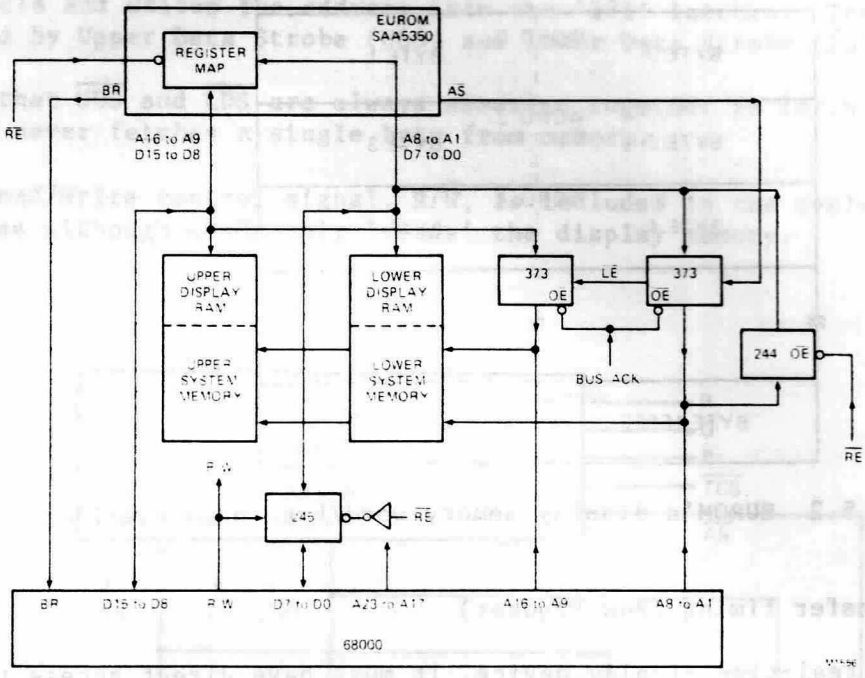


Figure 5.3 'Connected' 68000 interface

The main data and address paths used in a 'connected' 68000 interface are shown in Figure 5.3. The outputs of the '373' latches are only enabled when the 68000 has yielded the bus in response to Bus Request, BR. When the register map is accessed, data is transferred via the upper part of the bus, and the microprocessor's low-order address is passed to EUROM via the '244' buffer. Simultaneously, the '245' bidirectional buffer disables the signals from the low-order data bus of the 68000.

The '244' and '245' buffers may be omitted in a 16-bit 'write-only' configuration where the least significant data byte is interpreted by EUROM as an address. Here it will generally be necessary for the microprocessor to hold a (readable) master copy of EUROM's scroll map contents at some location in main memory.

If the possible 20 ns delay for accessing the Scroll Map is too long for the required speed of operation, a copy of the Scroll Map should be transferred to the main memory.

Microprocessor access to EUROM's register map

The set of internal registers comprising the register map, behave as an 8-bit wide RAM connected to the upper part of EUROM's bus; see Figure 5.4. The control signals $\overline{UDS}$ and R/$\overline{W}$ are reversed to become inputs and the register map is enabled with Register Enable, $\overline{RE}$. Addresses are input via the lower portion of the bus. A Data Transfer Acknowledge signal $\overline{DTACK}$, is also generated to indicate to the microprocessor that data transfer is complete.

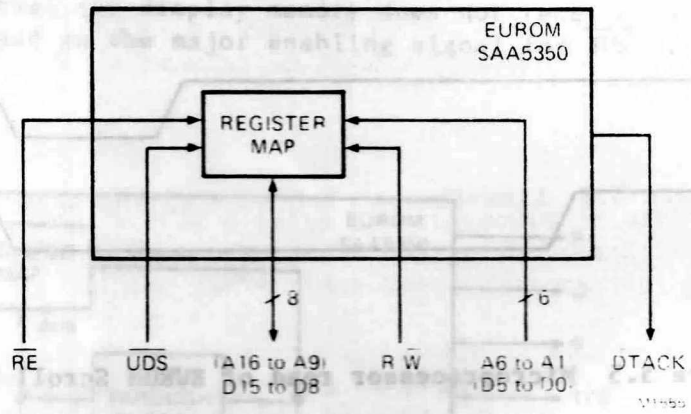


Figure 5.4 Processor access to EUROM's internal register map

Timing waveforms for microprocessor read and write operations are shown in Figures 5.5 and 5.6.

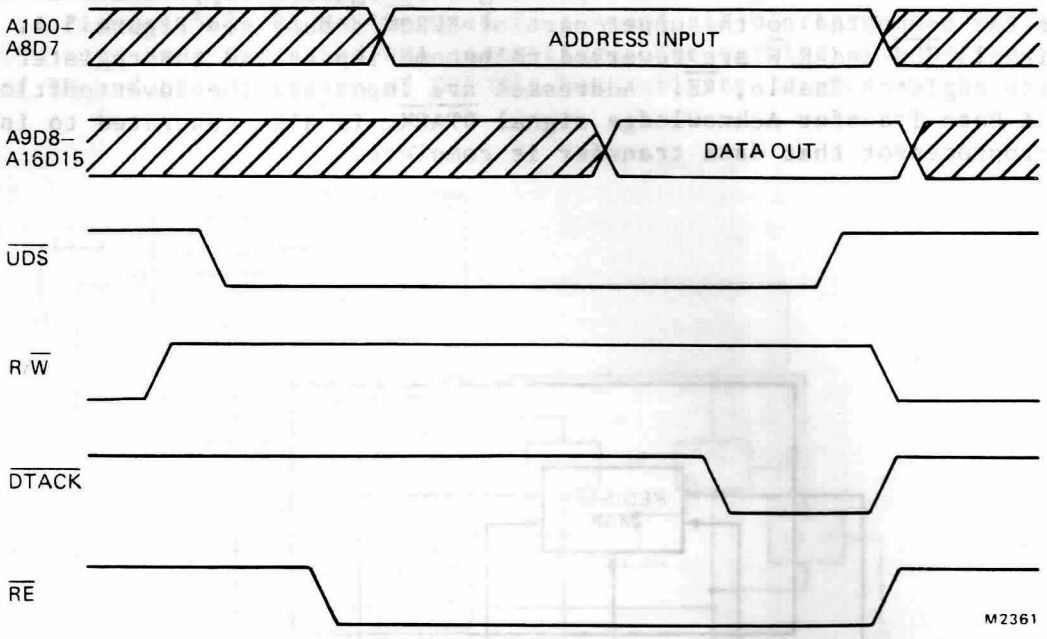


Figure 5.5 Microprocessor read of EUROM Scroll Map

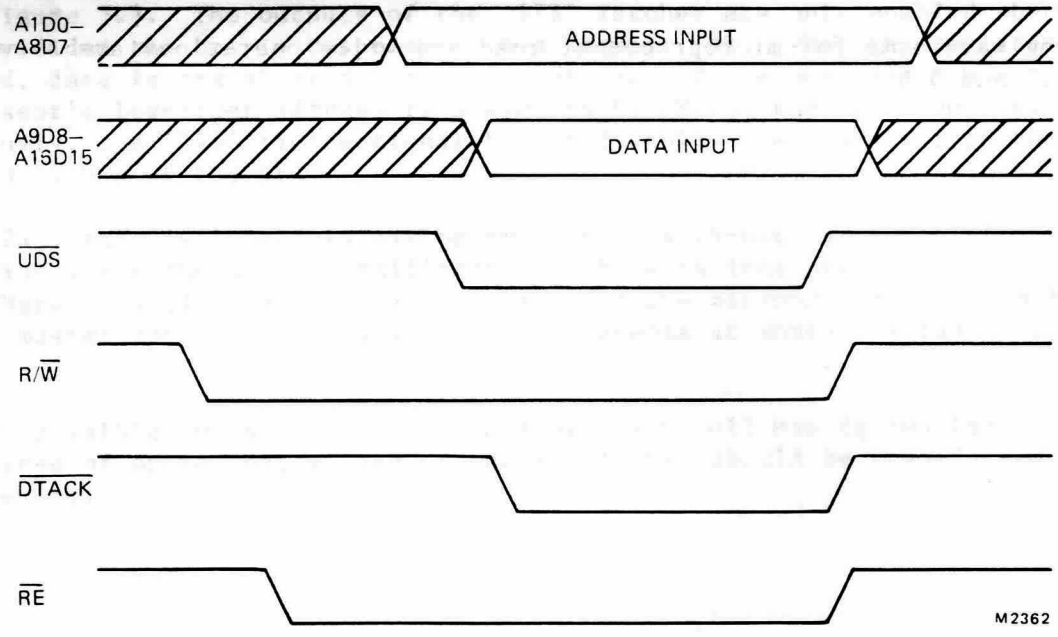


Figure 5.6 Microprocessor write to EUROM Scroll Map

8-bit microprocessor operation

Although the control bus is optimised for the 68000, EUROM will operate with a number of widely different industry-standard 8-bit and 16- (or more) bit microprocessors such as 80(1)88, 68008, 8051, etc. The interfacing of 8-bit microprocessors to the 16-bit wide display memory is simplified by EUROM's on-chip link-through buffer which provides the microprocessor with bidirectional access to the lower (odd byte) half of the display RAM. The link-through buffer is enabled with Buffer Enable, $\overline{BUFEN}$, and its Send/Receive direction is controlled by S/$\overline{R}$ which is physically the same EUROM pin as R/$\overline{W}$. Link-through buffer timing waveforms are shown in Figure 5.8.

Figure 5.7 shows the main data and address paths used in a 'connected' 8-bit microprocessor interface. Although this is very similar to the 68000 interface, it should be noted that the display memory does not receive A0 as an address, but A0, when high, is used as the major enabling signal for $\overline{BUFEN}$.

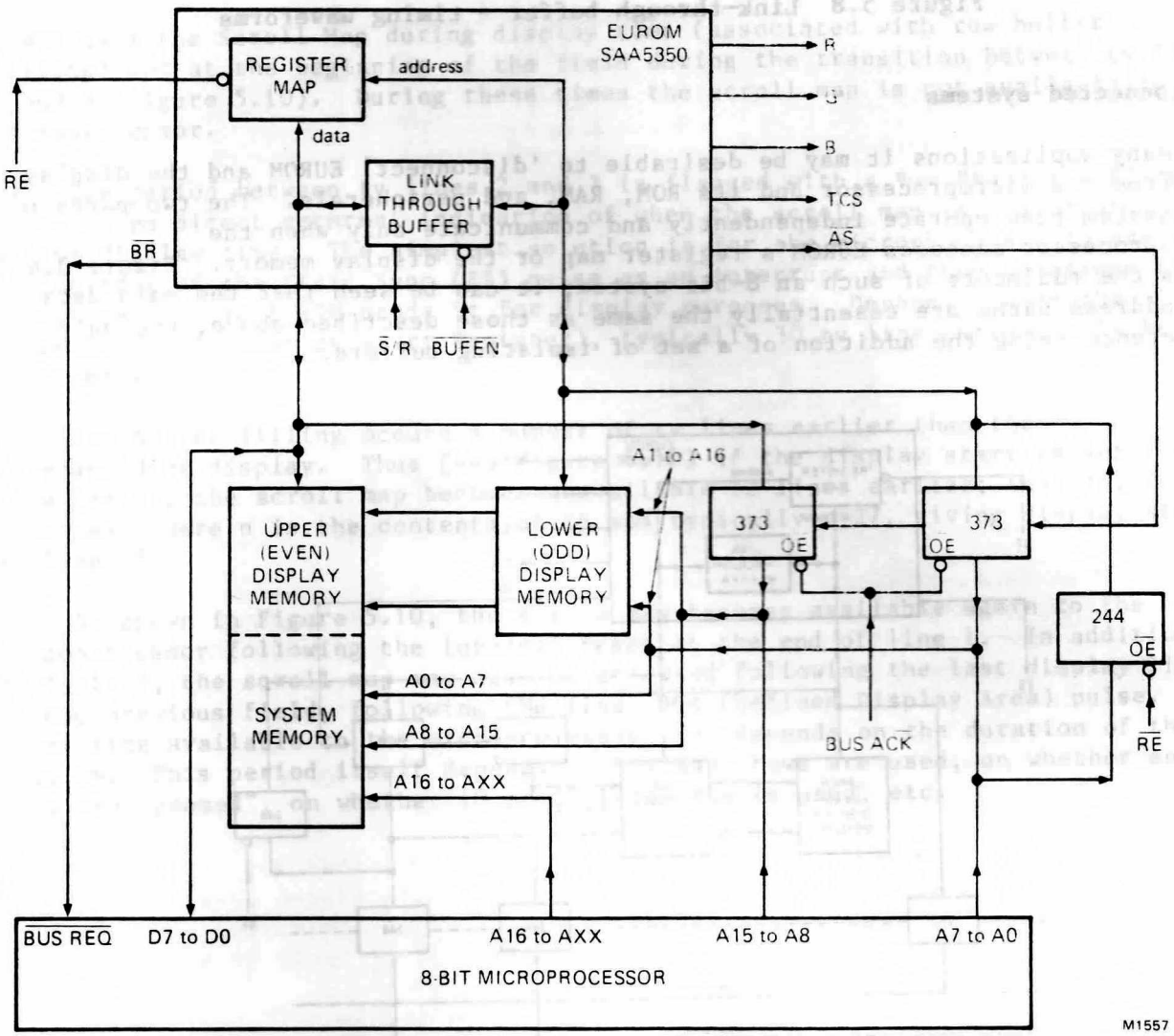


Figure 5.7 'Connected' 8-bit microprocessor interface

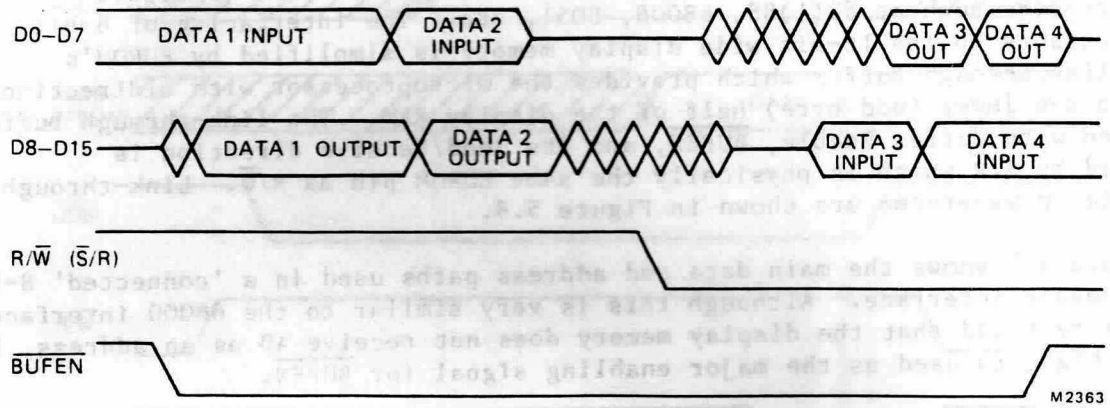


Figure 5.8 Link-through buffer - timing waveforms

Disconnected systems

For many applications it may be desirable to 'disconnect' EUROM and the display RAM from the microprocessor and its ROM, RAM, and peripherals. The two parts of the system then operate independently and communicate only when the microprocessor accesses EUROM's register map or the display memory. Figure 3.6 shows the rudiments of such an 8-bit system; it can be seen that the main data and address paths are essentially the same as those described above, the only difference being the addition of a set of isolating buffers.

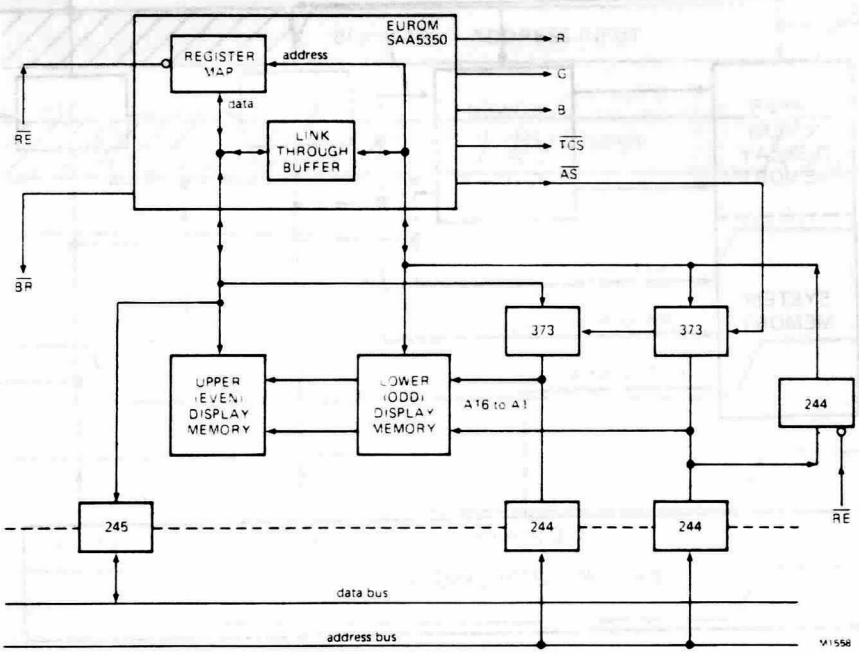


Figure 5.9 'Disconnected' 8-bit system

5.3 CYCLE INTERLEAVING

As described above, EUROM can be used in a number of microprocessor configurations. EUROM cycles memory at 500 ns and many microprocessors require approximately the same memory speed. By allocating alternate 250 ns memory cycles to EUROM and to the microprocessor - 'cycle interleaving' - the microprocessor can be made to appear to have almost completely free access to display memory.

One way of achieving this in practice is to synchronise events so that the microprocessor uses the 250 ns time slot immediately following the rising edge of EUROM's $\overline{UDS}$. EUROM facilitates this because there is a specified relationship between the 6 MHz output on the F1/F6 pin and the rising edge of $\overline{UDS}$. When EUROM is not using the bus at all (i.e. when $\overline{BR}$ is not aserted), the microprocessor can have completely free access to display memory without any need for wait states.

5.4 SCROLL MAP AVAILABILITY

EUROM uses the Scroll Map during display time (associated with row buffer filling) and at the beginning of the field during the transition between tv lines 2 and 3 (Figure 5.10). During these times the scroll map is not available to the microprocessor.

The period between tv lines 2 and 3 is flagged with a Bus Request (BR) but there is no direct external indication of when the scroll map is unavailable during display time. The simplest solution is for the microprocessor to use the rising edge of the Field Sync (FS) pulse as an interrupt and then to access the scroll map before EUROM needs it for display purposes. Depending upon the contents of R5 (Display Start Register), typically 30 tv line periods will be available.

Row buffer filling occurs a number of tv lines earlier than the corresponding display. Thus (see Figure 5.10) if the display start is set to line (n+15), the scroll map becomes unavailable 12 lines earlier; that is, from line n+3. Here n is the contents of R5 and typically n=27, giving display start at line 42.

As shown in Figure 5.10, the scroll map becomes available again to the microprocessor following the internal reset at the end of line 1. In addition, if desired, the scroll map may also be accessed following the last display line of the previous field, following the final DDA (Defined Display Area) pulse. The exact time available to the microprocessor then depends on the duration of the display. This period itself depends on how many rows are used, on whether any rows are "zoomed", on whether 10 or 12 lines/row is used, etc.

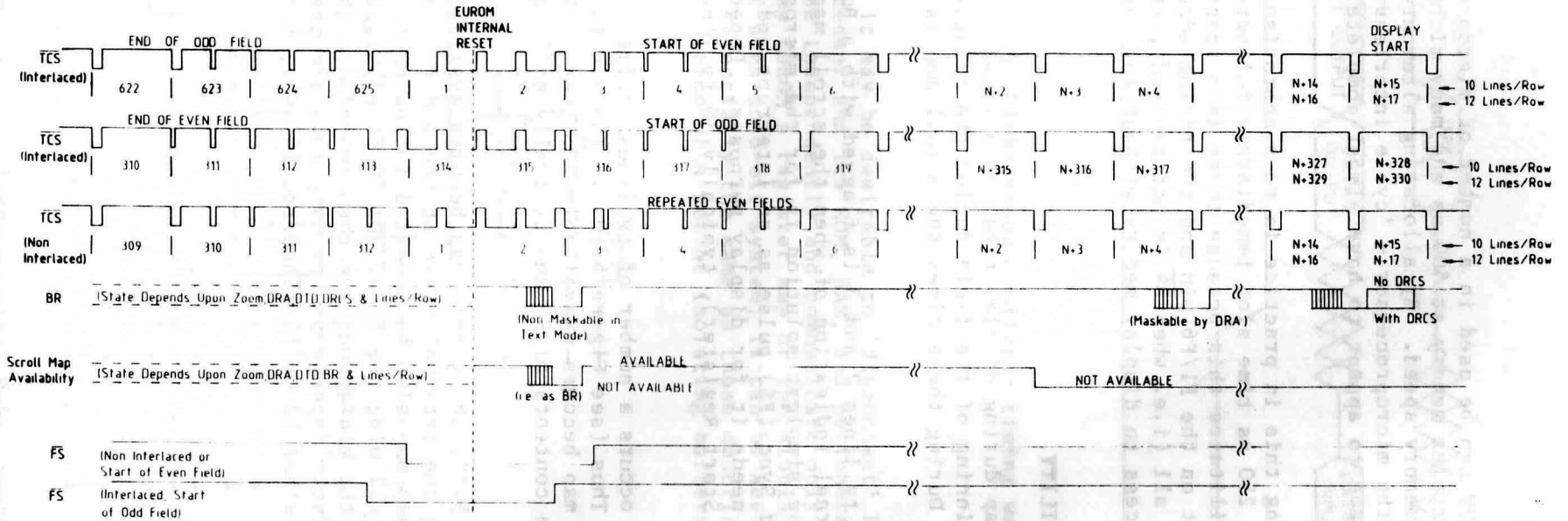


Figure 5.10 Field Timing Waveforms

5.5 SYNCHRONISING EUROM TO AN EXTERNAL VIDEO SOURCE

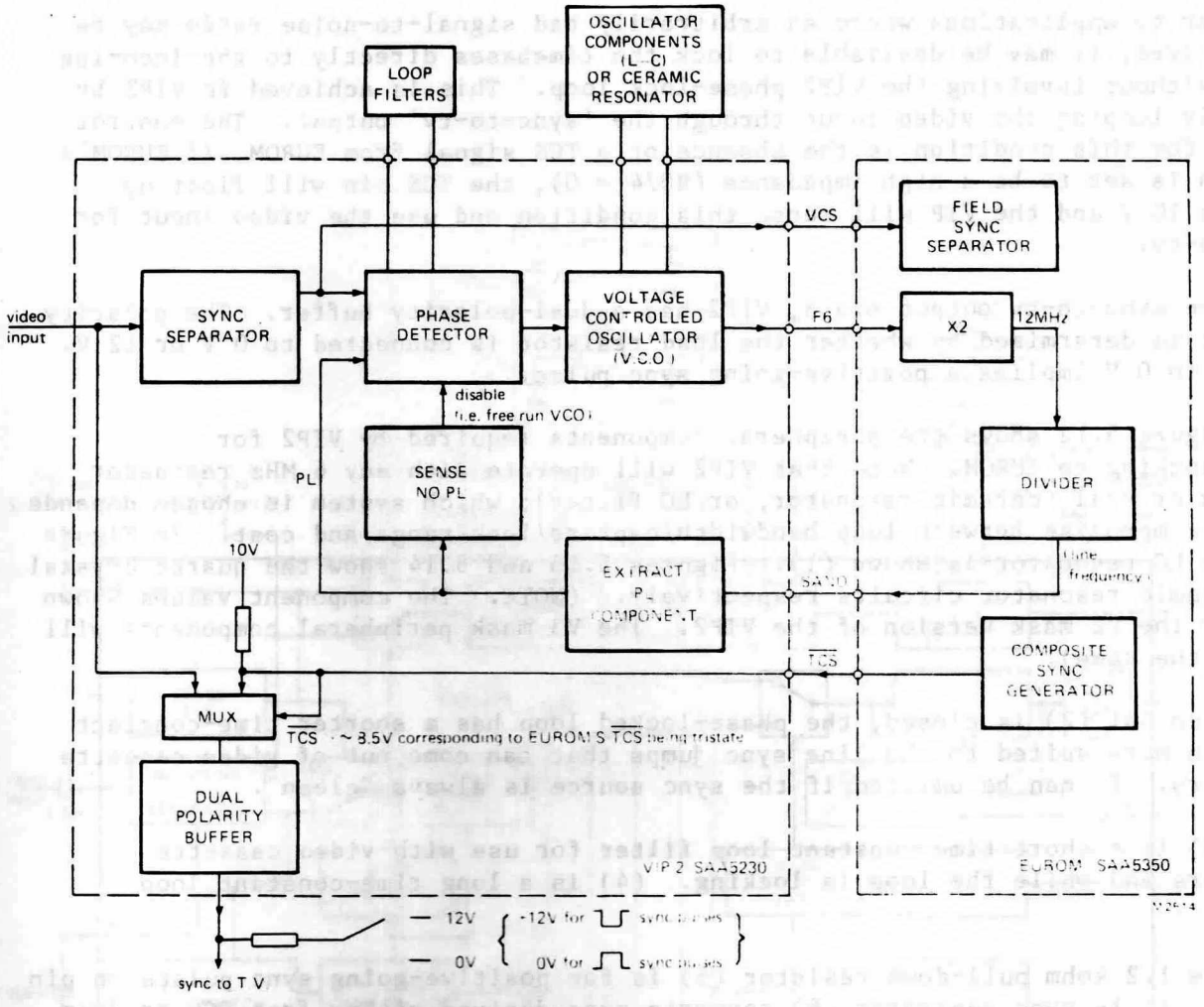


Figure 5.11 External Video and EUROM

Figure 5.11 shows how a signal from an external source of video, such as a video disc, is fed to EUROM. The incoming signal is first applied to the adaptive sync separator in VIP2 (SAA5230). Separated (composite) sync is then phase-compared in a line frequency phase detector with the PL (Phase Lock) component of SAND, a line frequency phase reference signal derived from EUROM's dividing chain. The error signal travels via the usual loop filters to the Voltage Controlled Oscillator (VCO), the output of which is fed to EUROM. This forms a horizontal phase-locked loop.

Vertically, composite sync is fed via the VCS (Video Composite Sync) pin to EUROM's field sync separator. When enabled, this signal resets EUROM's vertical dividing chain directly. It is also possible to feed such a system with a 525-line source of video but the field sync 'flywheel' effect will then be lost.

If the video disc player is stopped or disconnected, VIP2 will generally no longer 'see' a source of video (although some video disc players do generate syncs when the disc is removed). It is then desirable to allow the phase-locked loop to free-run and for EUROM to ignore its VCS input. This can be achieved by setting EUROM's register bit R0/2 = 1, thus disabling the PL component of SAND. VIP2 senses this and disables the loop phase detector, forcing the VCO to free-run.

Vertically, EUROM can be made to ignore the VCS input by setting register bit RO/1 = 0.

For tv applications where an arbitrarily bad signal-to-noise ratio may be encountered, it may be desirable to lock the timebases directly to the incoming video without involving the VIP2 phase-lock loop. This is achieved in VIP2 by directly looping the video input through the 'sync-to-tv' output. The control signal for this condition is the absence of a TCS signal from EUROM; if EUROM's TCS pin is set to be a high impedance (RO/4 = 0), the TCS pin will float up towards 10 V and the VIP will sense this condition and use the video input for sync-to-tv.

The sync-to-tv output pin 8, VIP2 has a dual-polarity buffer. The polarity of sync is determined by whether the load resistor is connected to 0 V or 12 V. A load to 0 V implies a positive-going sync pulse.

Figure 5.12 shows the peripheral components required by VIP2 for phase-locking to EUROM. Note that VIP2 will operate with any 6 MHz resonator (quartz crystal, ceramic resonator, or LC filter); which system is chosen depends on the compromise between loop bandwidth/capture/lock range and cost. In Figure 5.12 an LC resonator is shown (1). Figures 5.13 and 5.14 show the quartz crystal and ceramic resonator circuits respectively. (NOTE. The component values shown are for the V2 mask version of the VIP2. The V3 mask peripheral components will not be the same).

When SW1 (2) is closed, the phase-locked loop has a shorter time-constant which is more suited to the line sync jumps that can come out of video cassette recorders. It can be omitted if the sync source is always "clean".

(3) is a short time-constant loop filter for use with video cassette recorders and while the loop is locking. (4) is a long time-constant loop filter.

The 1.2 kohm pull-down resistor (5) is for positive-going sync pulses on pin 1. The simple sync separator (6) converts sync derived either from TCS or from the video input into a TTL-level composite sync waveform. The 5.6 kohm resistor (7) on pin 13 prevents spurious operation of the VIP2 data clock oscillator.

5.6 POWER-ON RESET

In general, EUROM does not use an internal power-on reset. The internal circuits are designed such that a valid state will occur after a number of clock periods without the need for any reset. There are, however, two exceptions:

a) At power-up, the VCS/OSCO pin is set to be an output so that if a crystal is attached, the oscillator will actually start.

b) At power-up, EUROM bus activity is inhibited until the DMA 'Warning Time' register, R2, is written to.

To re-trigger the internal power-on reset circuit, the VDD pin must be taken lower than about 2.4 V.

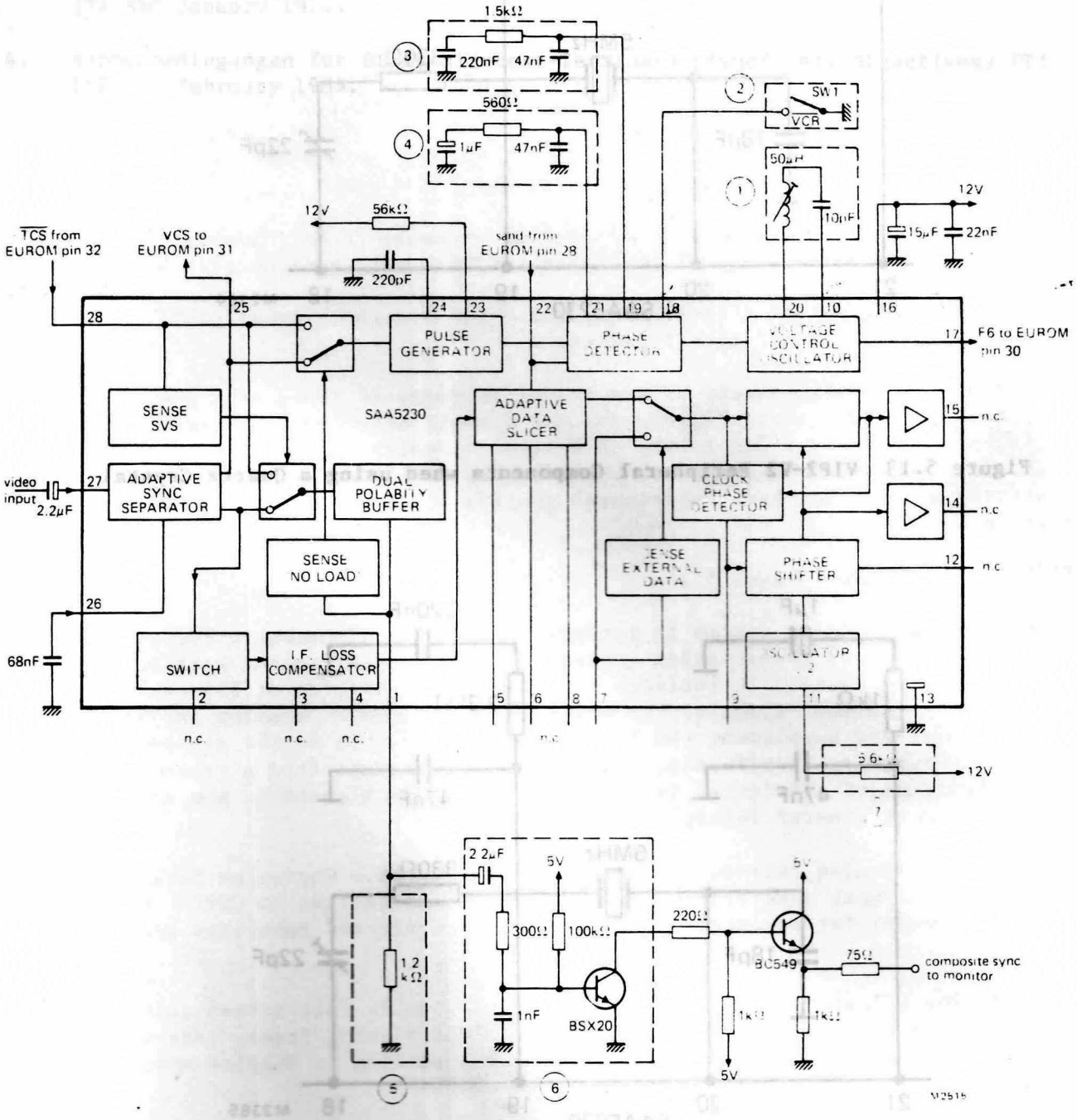


Figure 5.12 Using VIP2 to phase-lock EUROM to an External Video Input

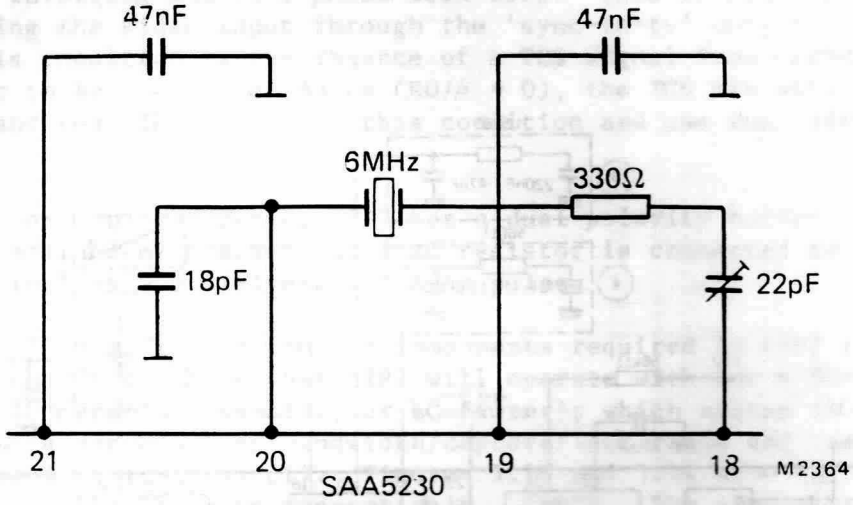


Figure 5.13 VIP2-V2 Peripheral Components when using a Quartz Crystal

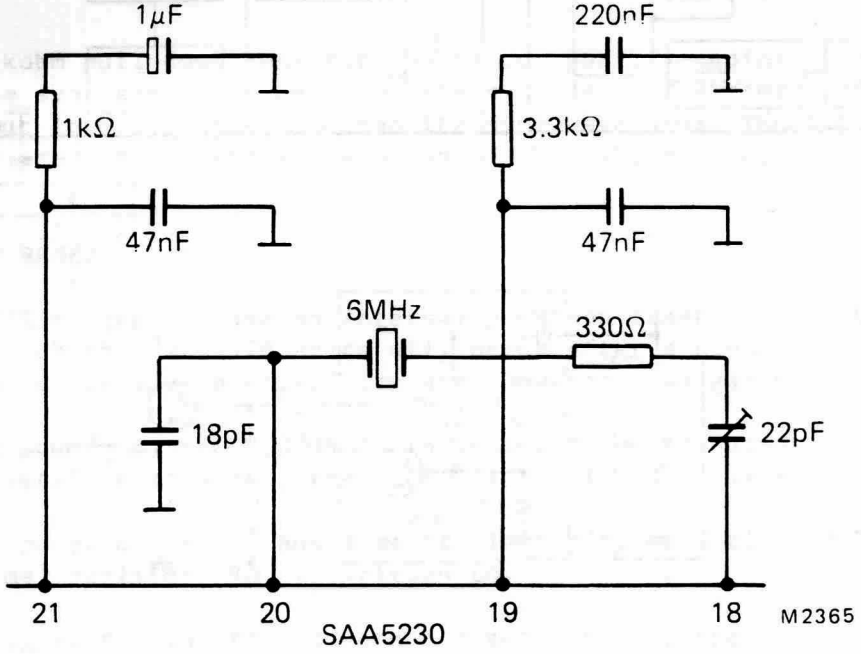


Figure 5.14 VIP2-V2 Peripheral Components when using a Ceramic Resonator

REFERENCES

1. Specification of a Basic Videotex Terminal operating to the European Videotex Service (2nd Edition). FTZ, Darmstadt, February 1982.

2. CEPT CD/SE Videotext Presentation Level Protocol (Revision of TCD 6-1) April 1983.

3. Specification of Television Standard for 625-line System-I Transmissions. ITA/BBC January 1971.

4. Rahmenbedingungen fur Bildschirmtext-Terminals (functional objectives) FTZ 157 D 2 February 1983.

APPENDIX 1

CEPT VIDEOTEX

1.1 CEPT VIDEOTEX

CEPT videotex is a compatible extension of the two original European videotex standards, including both serial spacing and parallel non-spacing attribute controls. It is standardised in 26 countries and permits full use of the existing databases. Its coding structure is sufficiently 'open' to allow for future expansion.

The standard incorporates the ISO character repertoire, a set of 335 characters which meet the requirements of the 40 Latin-based European languages. Also, an expanded range of attributes offers greater editorial flexibility to the information provider and better legibility for the viewer.

Videotex standards can be considered as existing at four downward-compatible levels of presentation protocol requiring progressively more display memory.

Basic alphamosaic system ('Level 1' in the World System Teletext (WST) specification)

This relatively simple system offers a limited range of characters (94 or 96) and block mosaics, eight standard colours, and a small selection of other attributes. It is typified by the UK Prestel videotex service and by the broadcast teletext services now in operation in twelve European countries. Its simplicity and economy have contributed greatly to the initial succes of these services.

'Alpha-DRCS' System ('Level 3')

The 'alpha-DRCS' coding system is compatible with the alphamosaic system in both display standard and transmission coding, but offers non-spacing attributes at the presentation level. It includes the full ISO repertoire of 335 characters, including mathematical symbols. In addition, 151 'smooth' mosaics and other line drawing characters supplement the block mosaics available in the alphamosaic system. Colours are redefinable, with up to 32 on-screen from a repertoire of 4096. The range of attributes includes underline, double-width, and double-size, which allow much greater editorial flexibility.

This system also introduces Dynamically Redefinable Character Sets (DRCS), potentially the most powerfull of the new techniques defined by CEPT. DRCS extends the character repertoire and permits sophisticated on-screen graphics to be displayed.

Additional features include the ability to specify full screen colours, extended flash (colour table flash and three-phase flash), freely intermixable double-height and double-width characters, and scrolling to display more information.

Alphageometric System ('Level 4')

This system of coding is pixel-rather than character-orientated and allows the transmission not only of alphanumerics but also of geometric primitives - vectors, arcs, polygons, etc.

Alphaphotographic System ('Level 5')

Alphaphotographic coding allows an analog picture of arbitrarily good quality to be transmitted using YUV coding (for example, from a video camera). Picture quality is limited only by the terminal resolution, transmission time, and amount of memory available. A practical terminal implemenation would include, say, alphamosaics coded to the 'alpha-DRCS' standard.

TOPIC	SECTION
	A
Address Paths	Sec 1.2, 5.2
Addressing Memory, Stack Mode	Sec 2.2
Addressing Memory, Explicit Fill Mode	Sec 2.3
Addressing Memory, 80 Characters/Row Mode	Sec 2.4, 3.5
Alpha-DRCS	Appendix 1
Alphageometric Videotex	Appendix 1
Alphamosaic Video	Appendix 1
Alphaphotographic Videotex	Appendix 1
AS	Sec 4.1, 5.1
Attributes, Addressing	Sec 2.2
Attributes, Enable/Default	Sec 4.7
	B
Background Colour	Sec 1.11, 4.13
Bit Map	Sec 1.2, 3.7
Black Screen Functions	Sec 4.6
Box/Window	Sec 1.11, 2.4, 4.7
Buffer Enable, BUFFED	Sec 5.2
Buffers, Isolating	Sec 5.2
Buffer, Link-through	Sec 5.2
Buffers, Row	Sec 1.2, 1.3, 1.7, 2.3, 2.4, 4.22, 5.4
Bus Request, BR	Sec 4.5, 5.1, 5.2, 5.3
	C
CEPT	Sec 1.1, 4.2, Appendix 1
Chapter Pointer	Sec 1.10, 4.7, 4.9, 4.10, 4.11
Chapter Addressing	Sec 2.2
Character Cells	Sec 3.1, 3.2
Character Generation	Sec 1.6
Character Size	Sec 1.11
Character Tables	Sec 1.6
Character/Row, 80	Sec 1.3
Character Table	Sec 1.9, 4.7
Clock	Sec 1.4
CLUT, DRCS	Sec 4.16
Colour, Background	Sec 1.11
Colour, Foreground	Sec 1.11
Colour Look-Up Table	Sec 1.9, 4.2
Colour Map	Sec 1.8, 1.9. 1.11, 3.2, 3.7, 4.2, 4.4
Colour Map Registers	Sec 4.17, 4.18, 4.19, 4.20
Colour, Screen	Sec 1.8, 1.11, 2.2, 2.3
Conceal, Character	Sec 1.11, 4.7
Conceal, Cursor	Sec 4.14
Connected/Disconnected System	Sec 5.2
Crystal	Sec 1.4, 1.5
Cursor	Sec 1.7, 1.8, 2.2, 4.12, 4.13, 4.14, 4.15
Cursor Flash	Sec 4.14
Cursor Registers	Sec 4.12, 4.13, 4.14, 4.15
Cycle Interleaving	Sec 5.2

TOPIC	SECTION
	D
Data Paths	Sec 1.2, 5.2
DCLUT	Sec 1.9, 4.16
Default Conditions	Sec 1.11
Defaults, Power Up	Sec 4.5, 4.7
Defaults, White Button	Sec 1.11, 4.7
Define Display Area, DDA	Sec 1.4, 1.7, 2.4, 4.3, 4.5, 4.22, 5.1, 5.4
Direct Fill	Sec 2.3
Disable Double Height, DDH	Sec 1.8, 4.22, 5.1
Disable RAM Access, DRA	Sec 1.8, 4.22, 5.1
Disable Text Display, DTD	Sec 1.8, 4.22, 5.1
Disconnected/Connected Systems	Sec 5.2
Display Area, Defined, DDA	Sec 1.4, 1.7, 2.4, 4.3, 4.5, 4.22, 5.1, 5.4
Display Attributes	Sec 1.10, 1.11
Display Command Registers	Sec 4.6, 4.7
Display Modes	Sec 4.7
Display Start Register	Sec 4.8, 5.4
Display Page Chapter Pointer	Sec 4.9
DMA	Sec 4.3, 5.2, 5.6
Dot Data Addressing	Sec 2.5, 3.1
Double Height Characters	Sec 1.8, 1.11, 4.22
Double Size Characters	Sec 1.11, 4.22
Double Width Characters	Sec 1.11, 4.22
DRCS	Sec 1.1, 1.3, 1.4, 1.6, 1.9, 1.10 2.1, 3.1, 3.3, 3.4, 3.5, 3.6, 3.7, 4.7, 4.16, 4.22
DRCS, Alpha	Appendix 1
DRCS Chapter Pointer A,B	Sec 4.10, 4.11
DRCS Character Table Bits	Sec 1.10, 3.3
DRCS-CLUT	Sec 1.9, 4.16
DTACK	Sec 4.1, 5.1, 5.2
Dynamic Display	Sec 1.11
	E
Enlarged Characters	Sec 1.3
Explicit Fill Mode	Sec 1.3, 1.11, 2.3, 3.7
External Memory	Sec 1.3
External Video	Sec 5.5
	F
Field Sync	Sec 1.5, 1.7
Flash Coding	Sec 2.2, 2.3, 4.6
Flashing, Default	Sec 4.7
Flash Modes	Sec 1.3, 1.11, 2.4
Flash Phases	Sec 1.11, 4.6
Field Sync, FS	Sec 4.3, 4.22, 5.1, 5.4
Foreground Colour	Sec 1.11, 4.13
Full Field DRCS	Sec 3.7
F1/6	Sec 4.3, 5.1, 5.3

TOPIC	SECTION
	I
Interfaces, Video	Sec 1.9
Interlace	Sec 4.3
Interleaving, Cycles	Sec 5.2
Invert, Foreground and Background	Sec 1.11, 4.7
ISO	Appendix 1
Isolating Buffers	Sec 5.2
	J
Jitter, Screen	Sec 1.5
	L
Lower Border Area, LBA	Sec 2.2
Lower Data Strobe, LDS	Sec 4.1, 5.1
Line Drawings	Sec 1.4
Line Drawing Symbols	Sec 1.6
Lines Per Row	Sec 1.4, 1.7, 2.4, 3.2, 4.3, 4.8, 5.4
Lining, Character	Sec 1.11, 4.7
Lining, Cursor	Sec 4.15
Lock Bit	Sec 2.2, 4.7
	M
Map, Bit	Sec 1.2, 3.7
Map, Colour	Sec 1.8, 1.9, 1.11, 3.2, 3.7, 4.2
Map, Register	Sec 4.1, 4.2
Mark-Space Ratio, Flash	Sec 1.11
Mathematics Systems	Sec 1.6
Memory, External	Sec 1.3, 2.2, 2.3, 2.4
Mode, Explicit Fill	Sec 1.11, 2.1, 3.7
Mode, Display	Sec 1.10, 2.1
Mode, Stack	Sec 1.11, 2.1, 2.2, 3.7, 4.22
Mode, 80 Characters/Row	Sec 1.11, 2.1, 2.4, 3.7
Mosaic Separation	Sec 1.11
Mosaics	Sec 1.4, 1.6
	N
Normal Size	Sec 4.7
Normal Text	Sec 4.6
	O
OSCO	Sec 4.3, 5.1, 5.6
Output Disable, OD	Sec 4.6
	P
Palette, Colour DRCS	Sec 4.6
Pinning of IC	Sec 5.1
PL, Phase Lock	Sec 4.3, 5.5
Pointer, Attributes	Sec 2.2
Pointer, Chapter	Sec 1.10
Power-on/Power-up Reset	Sec 5.6
Prestel	Appendix 1

TOPIC	SECTION
R	
RAM	Sec 1.3, 1.8, 4.22, 5.2
Read/Write, R/$\overline{W}$	Sec 4.1, 5.2
Register, Display Status	Sec 4.8, 5.4
Register Enable, RE	Sec 4.1, 5.2
Register Map	Sec 4.1, 4.2, 5.2
Reset, Power-Up	Sec 5.6
Reveal, Screen	Sec 4.7
RGB	Sec 1.9, 1.11, 4.6
ROM, On-chip	Sec 1.10
Row Attributes	Sec 1.8
Row Buffers	Sec 1.2, 1.3, 1.7, 2.3, 2.4, 4.22, 5.4
Row Data Registers	Sec 4.27
Row Reveal	Sec 1.8
Rows Per Page	Sec 4.8
Row Selection	Sec 1.4, 1.7
Row, Twenty fifth	Sec 1.4
Rules for Character Display	Sec 1.11
S	
Sandcastle	Sec 1.5, 4.3
Scan Test Mode	Sec 4.3
Screen Colour	Sec 1.8, 1.11, 2.2, 2.3
Screen Jitter	Sec 1.5
Screen Reveal	Sec 1.8
Scroll Map	Sec 1.4, 1.7, 1.8, 4.8, 4.22, 5.2, 5.4
Send/Receive, S/R	Sec 5.2
Size of Characters	Sec 1.11, 2.3
Slave, Phase-Locked, Sync	Sec 1.5
Slave, Simple, Sync	Sec 1.5
Stand-alone Sync	Sec 1.5
Stack Architecture	Sec 1.3
Stack Coding	Sec 1.3, 2.2
Stack Mode	Sec 1.11, 2.1, 2.2, 3.7, 4.12, 4.13, 4.14, 4.22
Superimpose Mode	Sec 4.6
Synchronising, External	Sec 5.5
Synchronising Modes	Sec 1.2, 1.5
Sync Register	Sec 4.3
T	
Table Bits	Sec 1.10, 3.3
Table, Character	Sec 1.6
TCS, Text Composite Sync	Sec 1.5
Text Display	Sec 1.8
Timing Chain	Sec 1.4
Timing, Transfer	Sec 5.2
Transfer Timing	Sec 5.2
TCS	Sec 4.3, 5.1, 5.5
Text Register	Sec 4.3
Text, Normal	Sec 4.6

TOPIC	SECTION
T	
TEXTEN	Sec 4.6
Translation Table, Row Address	Sec 1.7, 1.8
Transparent	Sec 1.11, 2.4, 4.2, 4.22
Transparent Background	Sec 4.7
TVEN	Sec 4.6
TV Mode	Sec 4.6
U	
Underline	Sec 2.4
Upper Data Strobe, UDS	Sec 4.1, 5.1, 5.3
Upper Border Area, UBA	Sec 2.2
V	
VCO	Sec 5.5
VCS	Sec 4.3, 5.1, 5.5, 5.6
VDS	Sec 4.6, 5.1
Video Input	Sec 5.5
Video Interfaces	Sec 1.9
Video Output	Sec 1.9, 5.1
Video Picture	Sec 1.11
Videotex Systems, CEPT	Appendix 1
W	
Warning Time, DMA	Sec 4.5, 5.6
WBEN	Sec 4.6
White Button, Defaults	Sec 1.11, 2.4, 4.2, 4.7
White Foreground	Sec 4.7
Window/Box	Sec 1.11, 2.4, 4.7
World System Teletext, WST	Appendix 1
Y	
YUV coding	Appendix 1
Z	
Zoom	Sec 1.7, 1.8, 4.22, 5.4
Numbers	
1 Bit Per Pixel/Dot	Sec 4.11
2 Bits Per Pixel/Dot	Sec 1.9
4 Bits Per Pixel/Dot	Sec 1.9
8 Bit Microprocessor Operation	Sec 5.2
10 Lines Per Row	Sec 2.4, 3.2, 4.3, 4.8, 5.4
12 Lines Per Row	Sec 1.7, 2.4, 3.2, 3.3, 3.4, 4.3, 4.8, 4.22, 5.4
40 Characters/Row Mode	Sec 2.3, 3.4
80 Characters/Row Mode	Sec 1.3, 1.11, 2.4, 3.7

Table 4.1 Register map (Registers 0 - 17)

	B7	B6	B5	B4	B3	B2	B1	B0	
0	VCS/$\overline{\text{OSCO}}$	F1/$\overline{6}$	FS/$\overline{\text{DDA}}$	TCS $\overline{\text{IN}}$/OUT	$\overline{10}$/12 LINES/ ROW	$\overline{\text{PL}}$ $\overline{\text{ENABLE}}$	ACCEPT VCS	INTER-LACE	SYNC REGISTER
1						T2	T1	T0	TEST REGISTER
2				T4	T3	T2	T1	T0	DMA "WARNING TIME" (T in uSec)
3	TV EN.	TEXT EN.	WBEN				PAL. (COL. DRCS)	FLASH INHIB.	DISPLAY COMMAND REGISTER 1
4				FULL FLD. DRCS	WHITE BUTTON	SCREEN REVEAL	M1	M0	DISPLAY COMMAND REGISTER 2 (Note 1.)
5			N5	N4	N3	N2	N1	N0	DISPLAY START (N IN LINES)
6			A16	A15	A14	A13	A12	A11	DISPLAY PAGE CHAPTER POINTER
7			A16	A15	A14	A13	A12	A11	DRCS CHAPTER 'A'
8			A16	A15	A14	A13	A12	A11	DRCS CHAPTER 'B'
9			C5	C4	C3	C2	C1	C0	CURSOR COLUMN
10	T2	T1	T0	F4	F3	F2	F1	F0	CURSOR CHARACTER CODE AND ATTRIBUTE REGISTERS
11		FLEN	C	B4	B3	B2	B1	B0	
12	U	P6	P5	P4	P3	P2	P1	P0	
13	A1	A0		D4	D3	D2	D1	D0	D-CLUT
14				A4	A3	A2	A1	A0	ADDRESS) COLOR PALETTE
15					D3	D2	D1	D0	DATA (R))
16					D3	D2	D1	D0	DATA (G))
17					D3	D2	D1	D0	DATA (B))

Table 4.2 Register map (Registers 18 - 58)

	B7	B6	B5	B4	B3	B2	B1	B0		
18									UNUSED	
(									)	
(									)	
(									)	
32									UNUSED	
R33	DIS--ABLE RAM ACCESS	DIS--ABLE TEXT DISP.	DIS--ABLE DOUBLE HEIGHT	R4	R3	R2	R1	R0	SCAN ROW 0	)
R(	"	"	"	"	"	"	"	"		)
R(	"	"	"	"	"	"	"	"		) SCROLL MAP
R57	DRA	DTD	DDH	R4	R3	R2	R1	R0	SCAN ROW 24	)
R58	DRA	1	X	X	X	X	X	X	SCAN ROW ROW '25'	)

Note (1) M0 and M1 determine the fundamental display mode:

M1	M0	MODE
0	0	Stack Mode
0	1	This Combination is Not Allowed
1	0	Explicit Fill Mode
1	1	80 Characters/Row Mode

Note (2) R = Readable Register (R33 to R58)

Note (3) Scan Row '25' corresponds to the Lower Board Area (LBA) at the bottom of the display.

Note (4) Individual register bits may be referred to as R register bit X. i.e. White Button is R4 bit 3.

Table 4.1 Register map (Registers 0 - 17)

Table 4.2 Register map (Registers 18 - 58)

	B7	B6	B5	B4	B3	B2	B1	B0	
0	VCS/ OSCO	F1/ 6	FS/ DDA	TCS IN/ OUT	10/12 LINES/ ROW	PL ENABLE	ACCEPT VCS	INTER-LACE	SYNC REGISTER
1						T2	T1	T0	TEST REGISTER
2				T4	T3	T2	T1	T0	DMA "WARNING TIME" (T in uSec)
3	TV EN.	TEXT EN.	WBEN				PAL. (COL. DRCS)	FLASH INHIB.	DISPLAY COMMAND REGISTER 1
4				FULL FLD. DRCS	WHITE BUTTON	SCREEN REVEAL	M1	M0	DISPLAY COMMAND REGISTER 2 (Note 1.)
5			N5	N4	N3	N2	N1	N0	DISPLAY START (N IN LINES)
6			A16	A15	A14	A13	A12	A11	DISPLAY PAGE CHAPTER POINTER
7			A16	A15	A14	A13	A12	A11	DRCS CHAPTER 'A'
8			A16	A15	A14	A13	A12	A11	DRCS CHAPTER 'B'
9			C5	C4	C3	C2	C1	C0	CURSOR COLUMN
10	T2	T1	T0	F4	F3	F2	F1	F0	CURSOR CHARACTER CODE AND ATTRIBUTE REGISTERS
11		FLEN	C	B4	B3	B2	B1	B0	
12	U	P6	P5	P4	P3	P2	P1	P0	
13	A1	A0		D4	D3	D2	D1	D0	D-CLUT
14				A4	A3	A2	A1	A0	ADDRESS) COLOR MAP
15					D3	D2	D1	D0	DATA (R))
16					D3	D2	D1	D0	DATA (G))
17					D3	D2	D1	D0	DATA (B))
18									UNUSED
(									)
(									)
(									)
32									UNUSED
R33	DIS--ABLE RAM ACCESS	DIS--ABLE TEXT DISP.	DIS--ABLE DOUBLE HEIGHT	R4	R3	R2	R1	R0	SCAN ROW 0) SCROLL MAP
R(	"	"	"	"	"	"	"	"	)
R(	"	"	"	"	"	"	"	"	)
R57	DRA	DTD	DDH	R4	R3	R2	R1	R0	SCAN ROW 24)
R58	DRA	1	X	X	X	X	X	X	SCAN ROW ROW '25')

Note (1) M0 and M1 determine the fundamental display mode:

M1	M0	MODE
0	0	Stack Mode
0	1	This Combination is Not Allowed
1	0	Explicit Fill Mode
1	1	80 Characters/Row Mode

Note (2) R = Readable Register (R33 to R58)

Note (3) Scan Row '25' corresponds to the Lower Board Area (LBA) at the bottom of the display.

Note (4) Individual register bits may be referred to as R register bit X. i.e. White Button is R4 bit 3.